Heuristic Approaches for Multi-objective, Multi Resource Constrained Project Scheduling Problem

Synopsis

Submitted by

Mr. Bibhuti Bhusan Tripathy

For the degree of

Doctor of Philosophy

(Faculty of Management)

Symbiosis International University

PUNE

In

April, 2013

Under the guidance of

Dr. L. Ganapathy

Professor

National Institute of Industrial Engineering

Vihar Lake, Mumbai, 400087

ABSTRACT

Resource Constrained Project Scheduling Problem (RCPSP) is modeled as a static and deterministic problem and one of the most classical problems in project scheduling problem with the objective of makespan minimization. It belongs to the class of the NP (non-polynomial) hard problems. It consists of set of activities with processing times and scarce resources per time. In RCPSP the objective to allocate the resources to a set of interrelated activities, with associated processing times and resources limited per time and precedence relations between activities. The aim is to find the minimal scheduling that respects the precedence relations and resource limits. The scheduling objective is to extend the project duration as small as possible to the original critical path duration in such a way that there resource constraints are met.

Heuristic based on priority rules have been one of the most important solution method for RCPSP with renewable resources. Priority rules based heuristic give some rule of thumb or experience to determine priority among activities for available of resources. They combine one or two priority rules and scheduling scheme (serial or parallel) to generate one or more makespan for the renewable resources.

This research consists of solution of different type of RCPSP. Firstly, it is aim to compare the heuristic algorithms and performance for medium and large sized project networks with resource constrained project scheduling problems and robust scheduling. The project with multi-mode activity execution aim is to minimize the project duration by choosing the activity priority and their execution mode. Secondly, this research proposes a genetic algorithm for finding an efficient optimal or near optimal algorithms to the RCPSP and need for a global search approached to the problem. Thirdly, the aim is to develop an algorithm for cost-time trade-off and resource availability in stochastic PERT network by crashing the project activities for minimization of duration with limited resources.

In this work heuristic based priority rule for the multi-resource with renewable resources are considered. We reviewed three types of heuristic rules: scheduling generation scheme, priority rule and mode selection rule for 90 randomly projects instant with 30 activities, 60 activities and 120 activities. All projects are analyzed by singe pass and multi pass heuristic base priority rules. The computation shows that parallel scheduling method performed better than serial scheduling method with the most of the priority rules tested. Also we have analyzed the effect of the problem parameters such as network complexity (NC), resource factor (RF) and resource strength (RS) on the overall performance in priority rules.

In this research, a genetic algorithm procedure has been proposed to produce an optimal and/or near optimal solutions. The permutation based elitist GA shows the performance of search for the optimal and/or suboptimal solution close to the lower bound solutions of RCPSP. The proposed one point crossover selection algorithm makes combination of the individual more randomly to produce best individual for next generation. This randomization process will increase the search space of the fitness function. Hence, the proposed algorithm is searched the fitness value in more wide space and converges between number of generation and standard deviation with a large space.

A stochastic PERT network with resource constrained model for construction project is presented. Various relationship, time-cost trade-off and resource constrained condition are considered in the model. We have generated a construction model having 40 activities with resource constrained. The proposed algorithm finds cost-duration relationship of the project. The algorithm is capable of handling a wide range of project sizes including large construction projects involving a large number of activities with resource constrained. The proposed model will help the decision-maker to provide a better understanding with the trade-off between the various components that affect the selection, cost, and resource allocation of projects. All computational results and conclusions and suggestions for future work are also reported for resource constrained project scheduling problem.

TABLE OF CONTENTS

CHAPTERS **Page**

Chapter 1 Introduction 7

1.1. The concept of Project 7

1.1.1. Concept of Project Management.....7

1.1.2. Project Scheduling.....8

1.2. Research Background 9

1.2.1. Heuristic in Project Management 10

1.3. Problem Statement.....11

1.4. Gaps in Existing Research.....12

1.5. Usefulness of Research.....12

1.6. Research Objectives.....13

1.7. Scope and methodology of the research.....13

1.7.1. Data source and Testing.....14

Chapter 2 Literature Review.....15

2.1 Literature Review.....15

Chapter 3 Heuristic Priority Rules Revisited.....20

3.1 Description of heuristics.....20

3.2 Scheduling Generation Scheme(SGS).....21

3.3 Priority Rules.....21

3.4 Test Design.....22

3.5 Experimental Investigation.....23

3.6 Single Pass Heuristic Priority Rules.....24

3.7 Multi-Pass Approach.....26

3.8 Forward-backward Scheduling Methods.....27

3.9 Parallel and Serial Method……..29

3.10 Effect of Problem Parameters on Performance of Priority Rule……..31

3.11 Robust Project Scheduling……..34

3.12 Computational Results of Robust Scheduling……..35

Chapter 4 Genetic Algorithms approach for Resource Constrained project Scheduling Problem……..42

4.1 Introduction of Genetic Algorithm……..42

4.2 Operator of Genetic Algorithm……..43

4.2.1 Initial Population……..44

4.2.2 Selection……..44

4.2.3 Crossover……..44

4.2.4 Mutation……..45

4.3 Genetic Algorithm for Resource Constrained Project Scheduling Problem……..45

4.3.1 Background Study……..45

4.3.2 Problem Definition and Formulation……..45

4.4 Proposed Genetic Algorithm for Resource Constrained Project Scheduling Problem……..48

4.5 Permutation based Encoding and Decoding……..49

4.5.1 Definition Phase……..49

4.5.2 Serial Scheduling Generation Scheme……..50

4.5.3 Elitism Selection……..56

4.5.4 One Point Crossover……..56

4.5.5 Proposed one point crossover algorithm……..57

4.5.6 Uniform Mutation Operation……..58

4.5.7 Termination Conditions……..59

4.5.8 Algorithm for Computing the Fitness Value of each Individual……..59

4.5.9 Experimental Result and Analysis……..60

4.5.10 Conclusions.....62

Chapter 5 Crashing PERT network with Resource Constrained conditions.....63

5.1 Introduction 63

5.2 Crashing.....63

5.3 Literature Review.....63

5.4 Problem Definition.....64

5.5 Mathematical Formulation.....65

5.6 Problem Analysis.....67

5.7 Proposed algorithm for crashing PERT networks with Resource Constrained.....72

5.8 Computational Result.....72

5.9 Summary and Conclusions.....74

Chapter 6 Conclusions and Future Research.....75

6.1 Summary of Research Conclusions.....75

6.2 Contributions of Research.....76

6.3 Recommendations for Future Research.....77

Chapter 7 Bibliography.....78

Chapter 1

1 Introduction

1.1 The Concept of Project

A project is defined as 'something projected or proposed for execution; a plan, scheme, purpose or a proposal'. A project is 'any undertaking with a defined starting point and defined objectives by which completion is identified'. According to Harrison, a project can be defined as a non-routine, non-repetitive, one-off undertaking, normally with discrete time, financial and technical performance goals. The project is a set of activities that must be completed in a pre-arrangement sequence.

1.1.1 Concept of Project Management

Project management is defined as the process used to develop a plan or blueprint to achieve the delivery of a new product or service. It is a process that usually requires the integration of complex steps to achieve the organizational goals. The critical path method (CPM) and Program Evaluation Review Technique (PERT) are the key tools for scheduling in project management.

In project management generally there is a due date for the project completion. Therefore, in some situations, a project should be completed in a shorter time than the normal time. The method of reducing the project duration by shortening an activity time at a cost is called crashing. This important issue in project management is the time and cost trade-off which can be modeled as a mathematical program. By assuming that the direct cost of an activity varies with time, the solution of these mathematical programs determines the optimal duration of a project so that project cost is minimized.

The CPM and PERT methods assume that appropriate resources are available when it needed. But in case of practice resources are constrained. Therefore, a variety of mathematical programs have been proposed to tackle so-called resource constrained scheduling problems. The heuristic methods and meta-heuristic methods are often used to solve large scale problems in practice.

1.1.2 Project Scheduling

Project scheduling has been a major objective of most models and methods proposed to help in planning and managing of projects. Initially, the study of project scheduling has been done considering just the duration and precedence conditions and ignoring the resource requirements. The most important methods to schedule a project assuming deterministic durations are CPM and PERT. However, most duration have a random nature and therefore, PERT is proposed to determine the distribution of the total duration.

The development of PERT and CPM has been widely used by practitioner for planning, coordinating and solving scheduling problems. The planning and scheduling of project activities has vital importance because of amount of time, resource and materials requirements will be known before the commencement of the project. The PERT and CPM is useful to generate information for the project manager to plan and control the project more effectively. CPM is useful only when the project deadline is fixed and the resources are not constrained by either availability or time. However, both methods assume availability of unlimited resources. A traditional CPM schedule is not realistic since it assumes unlimited resources, some of which are highly limited in practice. Therefore, resource scheduling arises as a result of project scheduling problem with availability of limited resources becomes the most important factor affecting the project scheduling performance. The scheduler must focus on making the best use of scarce resources through the project scheduling problem.

In project management, managing resources is a difficult process. The main task of the project scheduling process and the algorithms is to schedule the exact allocation of resources (e.g. people, machines, raw materials, etc.) to activities over time. A manger does not want to have too few resources because work will not get done in time and have too much of the resources, because the cost of ideal time cuts directly into profit. So the scheduler must take into account the trade-off between the reduction of certain activity duration by allocating more resources and increase in cost required as more resources are consumed. The common objective on a scheduling is the minimization of the project duration (makespan).

1.2 Research Background

Resource Constrained Project Scheduling Problem (RCPSP) is modeled as a static and deterministic problem and one of the most classical problems in project scheduling problem with the objective of makespan minimization. It consists of set of activities with processing times and scarce resources per time. In RCPSP the objective to allocate the resources to a set of interrelated activities, with associated processing times and resources limited per time and precedence relations between activities. The aim is to find the minimal scheduling that respects the precedence relations and resource limits (Brucker et al 1999). The scheduling objective is to extend the project duration as small as possible to the original critical path duration in such a way that there source constraints are met.

The activities are interrelated by two kinds of constraints, first, precedence constraints force each activity not to be started before its entire immediate predecessor activities comprised in the set have been finished. Second, performing the activities requires renewable resources. In all the cases resources may be of three categories: (1) renewable which means that resources are limited in quantity but renewable from period to period (2) Non-renewable means that both the total amount is limited over the project life (3) double constrained means that the both the total and pre-period resource amounts are limited. When the resources are renewable, activity duration are resource driven. The activity durations are independent continuous random variables, preemptive is not allowed. The problem is to determine when the activity will begin and which resource duration mode should be used in order to minimize the project duration. The objective of the problem is the feasible sequence for project activities such that makespan of project minimized.

The RCPSP is one of the most general and most difficult project scheduling problems and as generalization of the well know job shop problem. It belongs to the class of the NP (non-polynomial) hard problems (Blazemicz et al 1983). Scheduling of multi projects under resource constrained is extremely difficult computation. Kelly (1963) solved the resource constrained project scheduling problem (RCPSP) for optimal solution or heuristic solution.

An optimization technique involves mathematical programming approach, including integer programming, dynamic programming, branch and bound methods have been considered for an optimal solution. Wiest (1963) used IP to solve the project scheduling problem with limited resources in the shorting duration. Davis (1975) applied linear programming (LP) and dynamic programming to search an optimal solution. In order to obtain good scheduling for problems of practical size, in efficient time, heuristic approaches have been developed based on methods like priority rules, branch-bound algorithms and local search techniques. In particular, a branch-and-bound algorithm implemented by Demeulemeester and Herroelen (1992) has produced a good result.

1.2.1 Heuristic in Project Management

If resources are not sufficient available for the commencement of an activity, the activity is delayed for non-adequate of resources. The aim is to minimize the project duration caused by the resource shortage. It is required to assigned priorities to the activities which demand for the same resources and best combination of activity priorities is searched by analytic or heuristic algorithms.

The heuristic can be defined as the method that facilitates the process of finding the solution group by means of a simple rule. The limited resource project scheduling problem falls into a category of combinatorial problems.

Heuristic based on priority rules have been one of the most important solution method for resource RCPSP with renewable resources. Priority rules based heuristic give some rule of thumb or experience to determine priority among activities for available of resources. They combine one or two priority rules and scheduling scheme (serial or parallel) to generate one or more makespan for the renewable resources.

1.3 Problem Statement

The main objective of researchers on Resource Constrained Project Scheduling Problem (RCPSP) is to obtaining optimal or near optimal solution for resource allocation and project completion. This research aim is to develop effective algorithms for RCPSP that search feasible optimal and/or near optimal solution for the RCPSP with complex network and limited renewable resources. The study of heuristic algorithms David & Pall (1975), Davis E. W. (1975), Patterson & Huber (1974), Davis & Heidorn (1971) have shown that one heuristic which gives a good result for a project might not give that much successful result for another project. This is a greatest disadvantage of the heuristic rules perform well on one problem cannot give grantee to perform well in the other problem. Even most sophisticated heuristic algorithms cannot give guarantee in which particular heuristic or combination of heuristic will produced best results for a given problem. Even if not guarantee finding global optima and/or near optima, heuristic algorithms are used widely for the solution of RCPSP.

Meta-heuristic methods like genetic algorithm (GA) have been recognized as a powerful techniques based on the biological concept of survival of the fitness. This GA has been proven a most effective at obtaining near optimal solutions than the analytical techniques.

This thesis consists of solution of different type of RCPSP. Firstly, it is aim to compare the heuristic algorithms and performance for medium and large sized project networks with resource constrained project scheduling problems. The project with multi-mode activity execution aim is to minimize the project duration by choosing the activity priority and their execution mode. Secondly, the aim is to developing a genetic algorithm for finding an efficient optimal or near optimal algorithms to the RCPSP and need for a global search approached to the problem. Thirdly, the aim is to develop an algorithm for cost-time trade-off and resource availability in stochastic PERT network by crashing the project activities for minimization of duration with limited resources.

1.4 Gaps in Existing Research

The survey conducted among the researchers revel that there is a gap between project management practice and the development of project scheduling theory. Many project scheduling procedures published over the past several years in the literature have not used by practicing project schedulers. Studies reveal that managers often use information systems for project planning mainly for communication and representation, rather than for optimization. Most research efforts have concentrated on single project scheduling, while surveys indicate that multi-project settings often occur in practice.

Most resource constrained project scheduling research efforts have been made under the assumption that the project scheduling world is deterministic, while uncertainties during project execution are quite common.

Effective and efficient scheduling methods like meta-heuristics (genetic algorithms), stochastic project scheduling, NP-hard problem and proactive/reactive scheduling procedures are the crucial topics for finding optimal or near optimal solution for RCPSP. We have focused to generate new algorithms for RCPSP and finding optimal and/or near optimal solutions.

1.5 Usefulness of Research

Scheduling and sequencing is concerned with the optimal allocation of scarce resources to activities over time. The project scheduling problem involves the scheduling of project activities subject to precedence resource constraints. The practical application helps practitioners to plan and execute projects with multi project resource constrained environment within time and limited budget. The powerful meta-heuristics method i.e. genetic algorithm (GA) will help practitioners to uplift the performance of the project planning, scheduling and satisfying the various customer specifications. The crashing stochastic PERT network with resource constrained will help the practitioners to trade-off between resource availability and project duration for completion of project with the limited budget.

1.6 Research Objectives

Decision making is the most important function of the management. It is the quality of the management decisions which ultimately decides on the profitability and long growth of any business. Today's business has to live with a large number of constraints like shortages of raw materials, delay in deliveries from vendor, insufficient funds etc.

The aim of the research is to analyzing the heuristic models and their scheduling generation scheme, priority rules, meta-heuristics, genetic algorithm (GA) models for project scheduling problems under resource constrained environment. These techniques address some important practical problems: (1) scheduling with uncertain estimates on activity durations with resource constrained condition (2) integrated planning of scheduling and resources allocation (3) Trade-off between the project completion times with project budgets.

1.7 Scope and Methodology of the Research

Mathematical programming has been applied extensively for scheduling problems. Problems have been formulated using linear programming, integer programming, mixed-integer programming, goal programming and simulation. The use of these approaches has been limited because scheduling problems belong to the class of NP-complete problems. To overcome these difficulties, the scheduling problems are decomposed into a number of sub problems and proposed number of techniques to find optimal and/or near optimal solutions.

The main focus of this research is to development of heuristic and meta-heuristic (genetic algorithm) based search for solving optimal/near optimal solution to large and medium size resource constrained problem. In order to achieve the objective of this research, the scope of this research include review heuristic rules and their performance in resource constrained conditions and robust project scheduling, developing genetic algorithm for project scheduling problems with resource constrained environment for finding completion of the project with minimum makespan. Also, we focus to develop an algorithm for crashing stochastic PERT network with resource constrained condition. The optimal solution is trade-off between project completion duration and cost under availability of resources.

1.7.1 Data Source and Testing

In the heuristic base search, we have three kind of projects selection for assessing the performance of the priority rules. The projects are randomly selected from PSPLIB sets. A set of 30-activity (J30), 60-activity (J60), and 120-activity (J120) problems are considered having each 30 instances, respectively. Total 90 problems are selected with 33 terms and four types of resources. The verification and comparisons of the existing methods has been performed in rescon software algorithm and the collected data are computed with relative deviations. Hypothesis testing for comparing of two mean has been conducted. We performed non-parametric test, 2-tailed Wilcoxon matched pairs singed rank test for testing the hypothesis.

We have developed a genetic algorithm (GA) to generate the projects with resource constrained for finding optimal or near optimal solutions. The GA program is coded with Microsoft visual C++ version 6.0 compiler. In the crashing stochastic PERT network with resource constrained condition, we have written an algorithm with LINGO 11.0 software. The date is inserted according to the model for finding optimal solution with resource constrained problem. In addition, new solution techniques, more powerful heuristics algorithm, and the computational power of modern computers have enabled these approaches to be used on larger problems.

Chapter 2

2 Literature Review

Traditionally scheduling in project management is between planning and executing, taking decision on the specific resources, time for completion of activities and project cost. CPM was developed by Morgan Walker of Du Pont and James Kelly of USA in 1957 for the use in planning and scheduling of the construction of a chemical plant. PERT was developed by Naval Engineers of USA Navy in charge of Polaris sub-marine missile project in 1958. Euro working group on project management and scheduling was established to assess the state of art and to stimulate progress in theory and practice. CPM and PERT is widely used for planning and scheduling tools in the construction industry. It regenerates information for the project manager to plan and control the project more effectively and efficiently.

Kelfer and Verdini (1993) explained the expediting projects in PERT with stochastic time estimation. Sevkina and Hulya (1998) tried to analyze on project management using primal-dual relationship. Williams (1995) suggested how to estimate the PERT duration time. Cho and Yam (1997) developed method for “Uncertain Important Measure". It has been intended to explore some of the problems for crashing stochastic networks.Abbasi and Mukattash (2001) developed method for crashing PERT networks.

The CPM and PERT is a fundamental technique developed for project management assuming under the unlimited resource availability. In real life project activities must be schedule under the limited resource; such as limited crew sizes, limited equipment amounts and limited materials. The practical problem of allocating resources over the time in a variety of situation is a difficult task. The scheduler must take account of the trade-off between availability of resources and activity duration. To solve these problems scheduler use mathematical models but the primary disadvantage is that they could not solve bigger or complex problems. The scheduling of tasks and the allocation of resources in medium to large schedule projects is an extremely hard problems and challenges of project management due to its complexity. The

resource constrained project scheduling problem (RCPSP) is one of the most classical scheduling problems.

RCPSP is a NP-hard problem (Blazewicz, Lenstra, and RinooyKan, 1883). NP-hard problems are very difficult to solve realistic size problem and it necessary to use of heuristic methods. In RCPSP, the complex problem with an exact solution is very time consuming. The application and difficulties of the RCPSP are continuously attracts the attention of researcher and project managers. A vast amount of research and literature is addressed RCPSP solution using optimization or heuristic approaches (Demeulemeester and Herroelen1992), Kolisch and Hartmann, 1999).

Hastings (1972) formulated an algorithm for searching an optimal solution by creating a solution tree in which the nodes are represented a sub problem and each branch of the tree represented as a set of activities. Patterson and Huber (1974) proposed bounding technique with 0-1 programming to produce minimum duration scheduling for the RCPSP by examining the feasibility of the series of zero-one programming problems rather than solving zero-one problem optimally.

Coelho and Tavares (2003) suggested a new crossover operator for activity lists called late join function crossover. The late join function crossover constructs a new individual by “adoption the father solution and swapping each adjacent pair that is in reverse order in the mother”. Xu et al. (2007) developed several algorithms and provides estimates of the solution quality and computational burden of these heuristics using the PSLIB problem library (Kolisch and Sprecher, 1996). They compare the solution quality obtained from other algorithms found in the literature and finds that these procedures are competitive with the best algorithms found in the literature. Alcaraz and Maroto (2001) developed a genetic algorithms based on activity priority value encoding with scheduling mode. The results showed that the variant using an encoding with scheduling mode generates better solutions than the variant where the encoding does not include the scheduling mode. Hartmann (2002) developed a self-adapting genetic algorithm for project scheduling under resources constraints conditions.

Solving in heuristic methods based on the serial and parallel scheduling generation schemes procedures make feasible solution to handle practical resource constrain project scheduling problem. Many heuristic approaches have been proposed in RCPSP. Ulusoy and Ozdamar (1995) made a comparable analysis on projects through parallel methods using different priority rules with non-renewable resources. In parallel method algorithms it takes multiple project decision criteria (resource constraints, activity duration, slack, latest finish, etc). Lova et al. (2000) developed several algorithms based on backward and forward scheduling technique for improvement of multi-project feasible schedules. On their study, the improved feasible schedule is obtained from heuristic rule, such as maximum total work content (MAXTWK), LFT and others.

Patterson (1976) described project scheduling and the effect of problem structure on heuristics performance. Ellen and John (1993) examined resource policies in a dynamic multi project environment. Abbasi and Arabiat (2001) analyzed project cash flow was classified according to a priority rule that can be described as a combination of the latest start time(LST) and shortest processing time(SPT) methods. Klevin (2000) used parallel methods and different priority rules and figured out that LFT priority gives better performance. Davis and Patterson (1975) looked at the effectiveness of various priority rules. In their study, the MINSLK priority rule gives the best result in comparison to other priority rules.

Talbot (1982) formulated the non-preemptive case with on resource constraints project scheduling problem for discrete duration resource tradeoff. He developed a solution technique that can be used for real life large scale problem. Drexl and Grunewald (1993) developed a stochastic model for solving RCPCP and it out forms the other existing heuristic rules. Lawrence and Morton (1993) addressed some work relating the single-mode, resource constrained multi-project scheduling problem.

Herroelen and Leus (2004) have developed the hypothesis of unrestricted resource availability by using a resource flow network for robust resource allocation to a feasibility baseline scheduling. A branch-and-bound algorithm was proposed by the authors for this purpose. Van

de Vander et al. (2005) addressed the issues of achieving a tradeoff between the quality robustness and solution robustness for resource unconstrained project scheduling.

There are meta-heuristic approaches like genetic algorithms, tabu search, simulated annealing and ant colony systems are describe for the solution of RCPSP. Genetic algorithms are stochastic search based upon the mechanism of natural selection and population genetics. Holland (1975) was developed genetic algorithms (GA) based on mechanism of natural selection in biological system. It uses random direct search by the process of genetic evolution and principle of "survival of fittest". Frank (2001) showed that the GA performed slightly better than tabu search (TS) procedure but required more computation time.

Hartmann (1998) developed a permutation based genetic algorithm making use of activity representation. The genetic algorithm of Hartmann (1998) performed best among the 16 heuristic tested in the computational study of Hartmann and Kolisch (2000). They used the serial SGS and add a gene that decides whether the activity list is scheduled in forward and backward direction. Leu and Yang (1999) proposed a multi-criteria computational model optimal scheduling model that integrated a time-cost trade off model, resource-limited model and a resource leveling model. Kohlmorgen et al. (1999) summarized their experiences with an implementation of a genetic algorithm for the RCPSP on parallel processors. The algorithms include parallel version of island model of genetic algorithms and of variants of the neighborhood model called diffusion model.

Leon and Ramamoorthy (1995), Ozdamar (1999), and Hartmann (2002, 1998) developed a genetic algorithm which was embedded by priority rules. Kolisch (1995) and Drexl (1996) integrated the priority rules with sampling schemes. Boctor (1990) and Li and Willis (1992) discussed the forward and backward scheduling technique based on the use of multiple priority rules within a single heuristic procedure.

Hartmann (2001) presented a genetic algorithm for scheduling projects of multiple models of activity execution. The genetic encoding is based on a precedence feasible list of activities and a

mode assignment. Valls et al. (2003) introduce a new a heuristic algorithm for RCPSP. The algorithm is implementation of fundamentals concepts of tabu search without explicitly using memory structures embedded in a population-based framework. Baar et al. (1998) developed two tabu search approaches. Lee and Kim (1996) compared a genetic algorithm (GA), a simulated annealing heuristic, and a tabu search method. Al Fawzan and Haouari (2005) developed a multi-objective tabu search heuristic for solving a bi-objective RCPSP. They consider the objectiveness of quality-robustness maximization along with makespan minimization and used several variants of the algorithms in order to find an approximate set of effective solution. Bozejko and Wodecki (2004) applied the parallel tabu search method to solve the parallel permutation scheduling problem.

Tabu search starts with looking to local minima. The search records moves in one or more tube lists in order to avoid retracing the steps used. Thomas and Salhi 1998, Klein 2000, Nonobe and Ibaraki 2002, Artigues et al. (2003) applied tabu search in resource constrained project scheduling problem. Dorigo and Gambardella (1997) employed ant colony optimization (ACO) imitates an ant movement to find an optimal solution. The main objective of ant system include positive feedback for discovery of good solutions computation avoid premature convergence, use of constructive heuristic and help acceptable solution in early stage of the search process. Based on the simulation result, the ant colony optimization (ACO) is robust and can be used in RCPSP for optimization solution. Hindi (2004) presented a solution scheme based on variable neighborhood search for solving RCPSP.

Chapter 3

3 Heuristic Priority Rules Revisited

3.1 Description of Heuristics

The heuristic method can be defined as the method that facilitates the process of finding the solution by means of a rule (Demeulemeester and Herroelen 1992). The heuristic method is one of the scheduling methods used for the resource constrained projects. The limited resource project scheduling problem falls under combinatorial problems. The limitations of optimization based techniques are not successful on these combinatorial problems; several heuristic scheduling models are developed to produce near optimal or good solutions. Heuristic priority rules have been one of the most important solution technologies for multi-resource constrained project scheduling problems. The heuristic method is used to assign rule priorities the jobs that competing for the renewable resources.

The heuristic procedures broadly fall into two categories, constructive heuristics and improvement heuristic. The major components of constructive heuristic are scheduling generation scheme and priority rules. The solution quality obtained from heuristic method is not guarantee optimal solution but rather produce a good feasible solution. The heuristic procedures for resource constrained project scheduling is used a priority to rank the activities and then activities with resource limitation. In heuristic method, a forward pass is regularly checked whether the resource demand is exactly the limits of available of resources. Priority rule based methods consist of two components: one or (more) scheduling schemes and one or (more) priority rules. The two different schemes for the generation of feasible schedule can be distinguished i.e. the serial scheme and parallel scheme. The scheduling scheme determines how to schedule is constructed. In each stage the generation scheme forms the set of all schedule activities, called the decision set. A special priority rule is then employed in order to choose one or more activities from the decision set which is schedule (Lova and Tormos 2001).

3.2 Scheduling Generation Scheme (SGS)

The serial Scheduling Generation Scheme (S-SGS) was developed by (Kelly, 1963) for resource constrained project scheduling problem (RCPSP). The activities are started the numbering in such a way that no one gets the lower number than any one of its predecessors. It is an activity oriented scheme. Each one activity, assigned mode is selected and scheduled at the earliest precedence and resource feasible computation time. There are two disjoint activities sets are associated: the schedule set that includes the activities already scheduled and the decision set that consist of the activities eligible for scheduling.

In the each stage an activity is selected by a priority rule and mode of selection rule. The activity is scheduled take the precedence relationships of the activities and capacity of the resources. The activities selected is removed from the decision set and added to schedule set. The same procedure is updated with the immediate successor of the activity. The S-SGS finished when all activities are in schedule set. The two methods serial scheduling and parallel method are used in this research.

3.3 Priority Rules

Project algorithm consists two phases. First phase, the project's CPM solution is programmed without any kind of resource limitation. In the second phase, the priority is used by activities are calculated. The activities are programmed according to the priority.

A priority rule is a mapping which assigns each activity in the decision set a value and an objective stating whether the activity with the minimum or the maximum value is selected. Priority rules based heuristics combine priority rule and schedule generation scheme (SGS) to construct a specific algorithm. Many heuristics priority rules have been developed for resource constrained project scheduling problem. The priority rules give different results depending on the number of resources, quantity of limitation and size of the project. The most well-known priority rules are Minimum Job Slack, Minimum Late Finish Time, Shortest Imminent Operation, Select Jobs Randomly etc.

3.4 Test Design

To compare the effect of the priority rules on the project duration, the projects are scheduled in a limitless resource condition. The lower bound of project duration is calculated by the program. After that, the projects are scheduled in resource constrained conditions according to the priority rules. The program looks into daily resource requirements and checked each activity resource requirement on a daily basis. If the required resource is more than the availability then the program delayed the activity to the following day. The process continued until the calculation of the finish time of the last activity.

A set of instance is necessary in order to compare the performance of the heuristic based priority rules in resource constrained project scheduling problem (RCPSP). The projects in these instance sets consist of 30, 60, and 120 activities, respectively. Each set has been generated using a full factorial design of parameters, which determine the characteristics of the resource and precedence constraints (Kolisch et al. 1995, Kolisch and Sprecher 1996). A set of 30-activity (J30), 60-activity (J60), and 120-activity (J120) problem instances consists of each 30 instances, respectively. In total 90 problems, the test design with 30 instances of 30 activities, 30 instances of 60 activities, and 30 instances of 120 activities are examined. Each activity has a single execution mode with fixed integer duration between 1 to 20 time units. The small size problems have 30 activities and large size problems have 120 activities. The project instances used four renewable resources with fixed availability (R_k) are assigned to each activity.

Each activity in the project has a known duration and predetermined level of resources usage over its duration. An activity can not start until all predecessor activities have finished. An activity once started is not interrupted. These instance sets obtained from the PSPLIB is used in this experiment. In http://129.187.106.231/psplib/dataob.html a set of test problem is available.

3.5 Experimental Investigation

To study the performance of the scheduling, the procedure parameters are characterized as follows: the priority rule (PR), scheduling generation scheme (SGS) and mode selection rule. The problem parameters are Network complexity (NC), Resource factor (RF) and Resource strength (RS). Priority rules are chosen according to the studies of Davis and Patterson (1975), Alvarez-Valdes and Tamarit (1989a), Vails et al. (1992), Uluzoy and Ozdamar (1989), and Boctor (1990).

The instances obtained from the PSPLIB are systematically generated by the problem parameters are characterized by Rainer Kolisch et al., (1995) and Rainer Kolisch (1996). The three problem parameters are network complexity (NC), resource factor (RS), and resource strength (RS). The network complexity reflects the average number of immediate successors of an activity. The network complexity (NC) is the ratio of non-redundant precedence relations to the number of activities. The complexity is defined as the ratio between number of arcs and numbers of node. The complexity of the projects is taken as 1.50, 1.80 and 2.10.

Table 1
Minimum project duration

Priority rules	Extr.	Priority value P_j^*
Activity Number (AN)	min	j
Shortest Processing Time (SPT)	min	d_j
Longest Processing Time (LPT)	max	d_j
Earliest finish Time (EFT)	min	EFT_j
Earliest start Time (EST)	min	EST_j
Latest Start Time (LST)	min	LST_j
Latest Finish Time (LFT)	min	LFT_j
Minimum total Slack (MINSLK)	min	LST_j-EST_j

*All parameters are calculated using shortest duration mode for each activity

Table 2

Problem parameters		
NC	**RF**	**RS**-
(1.50, 1.80, 2.10)	(0.25, 0.50, 0.75, 1.00)	(0.30, 0.50, 0.70)

Heuristic based priority rule for RCPSP with resources has three components: A SGS, a priority rule and mode selection rule. This is based on the assumption that the computational effort for constructing one schedule is similar in most heuristics. This holds in particular for methods which apply the serial SGS or parallel SGS. We have tested with all SGS with forward and backward mode with a priority rule. Table 1 provides an overview of the eight priority rules. In the first column, the name of the each rule is display. In the second column, extrema select maxima or minima and in the third column priority value (P_j) parameter are displayed.

The main evaluation criteria adopted is the average deviation with respect to the lower bound that supposes the critical path length. The computational experience has been performed on a pentium-based personal computer with 600 MHZ and 120 RAM. Boctor (1990) and Vails et al. (1992) considered the serial scheduling scheme(S-SGS) and the parallel scheduling scheme (P-SGS). In this study, we have taken both serial and parallel scheduling problems with resource constrained and the problem parameters.

3.6 Single Pass Heuristic Based Priority Rules

Single-pass priority rule based heuristics employ one SGS and one priority rule in order to obtain a feasible schedule. The priority rules are used to determine the order in which the activities of D will be selected for scheduling. Each test problems are randomly selected in the PSPLIB library. The 90 test problems are solved with each of the 33 tandems using heuristic algorithms. The mean and standard deviation from the lower bound i.e. optimal objective function value is calculated for each of the 8 priority rules. Table 3 show the performance of priority ruled tested

in S-SGS scheduling scheme. The priority rules are written with respect to ascending mean. Also Table 3 shows the average percentage increasing of project duration over lower bound (critical path length) and the dependent on the number of resource type involved.

The result shows that LST, LFT, MINSLK, EFT seems to be better performing rules in serial generation scheme. The performance and ranking of the classical rules (Davis and Patterson 1975, Alvarez-Valdes and Tamarit 1989a) is comparable to the result of our studies. It seems that using the LST is the most appropriate rule to minimize project duration. The heuristic tandems are consistently best of all 90 problems based on the average percentage increasing of project duration, four heuristic LST, LFT, MINSLK, EFT seems to be better performing rules. Their average percentage increase over the critical path length is 32.57%, 33.02%, 40.98% and 41.94% respectively. The rules described in table 3 have been tested with static approach that is, the priority values are calculated only once before starting the scheduling process. The feasible schedule is very fast.

Table 3: Performance of priority rules with serial single pass

Priority rules	**Mean**	**Deviation**
LST	**32.57**	**30.53**
LFT	**33.02**	**30.57**
EST	**40.98**	**34.72**
MINSLK	41.94	33.90
EFT	46.05	35.65
LPT	46.51	35.90
RANDOM	53.75	37.33
SPT	56.76	40.51
SIO	57.17	40.80

The combination of 8 priority rules with parallel-SGS is consider for analyze the performance of single pass heuristic priority rules. If ties occur, the lower activity number is considered. The result is show in table 4.

Same 90 test problems are solved with each of the 33 tandems using heuristic algorithms with parallel single pass method. Table 4 show the performance of priority ruled tested in P-SGS scheduling scheme. Table 4 shows the average percentage increasing of project duration over lower bound (critical path length) and the dependent on the number of resource type involved.

In the P-SGS scheduling scheme, based on the average percentage increasing of project duration, four heuristic LST, LFT, MINSLK, EST are performing better of the priority rules. Their average percentage increase over the critical path length is 30.33%, 30.92%, 36.53% and 38.01% respectively. The priority rule with the lowest average deviation is LST, LFT and MINSLK gives better performance. The remaining priority rules have significantly worst result than the other priority rules.

Table 4: Performance of priority rules with parallel single pass

Priority rules	**Mean**	**Standard Deviation**
LST	**30.33**	**29.40**
LFT	**30.92**	**29.72**
MINSLK	**36.53**	**31.63**
EST	38.01	32.88
EFT	38.94	33.29
SPT	41.74	33.17
RANDOM	43.66	35.08
LPT	43.98	34.73

3.7 Multi-Pass Approach

There are many possibilities to combine scheduling generation scheme (SGS) and the priority rules into a multi-pass approach. The most common multi-pass methods are: multi- priority rules methods, forward-backward scheduling methods and sampling methods (Hartmann and Kolisch 2000). Each combination finds different algorithms and reported as solution with feasible lowest completion time. In multi-priority rule different priority rule is used and the best schedule is selected (Boctor 1990). Forward-backward scheduling methods are the basic of the new technology proposed by Boctor (1996).

3.8 Forward-backward Scheduling Methods

The forward sequence is completely a partial sequence by scheduling each activity as early as possible. Forward-backward scheduling methods use a scheduling generation scheme (SGS) to iteratively schedule the project by alternatively between forward and backward scheduling. This multi-pass heuristic scheduling approach was explained by Li and Willis (1992) and used by Ozdamar and Ulusoy (1996), Lova et al.(2000) and Thomos and Lova (2001).

In this research work, analysis has done for multi-pass heuristic rules by forward-backward scheduling methods. In table 5, the performance of priority rules in serial backward and forward pass with respect to the lower bound are examine for different combination of SGS. The lowest average deviation with respect to the lower bound with the combination of S-SGS is LFT with 30.84% shows minimum from best three priority rules. Other two LST and EST give the best result compare to other heuristic rules. The average deviation with respect to lower boundary is 31.53% and 35.76%. Table 6 shows the performance of priority rules in parallel backward and forward pass method. The lowest average deviation with respect to the lower bound with the combination of P-SGS is LFT (29.09%) shows minimum from best three priority rules. Other two heuristic rules LST and MINSLK give average deviation 29.39% and 33.42% respectively.

Table 5: Performance of priority rules in serial backward-forward pass

Priority Rules	**Mean Forward**	**Mean Backward**	**Average BF**
LFT	33.02	28.67	**30.84**
LST	32.57	30.48	**31.53**
EST	40.98	30.54	**35.76**
EFT	46.05	30.59	38.32
MINSLK	41.94	35.36	38.65
LPT	46.51	35.22	40.86
RANDOM	53.75	35.79	44.77
SPT	56.76	36.33	46.55

Table 6: Performance of priority rules in parallel backward and forward pass

Priority rules	Mean Forward	Mean Backward	Average BF
LFT	30.92	27.25	**29.09**
LST	30.33	28.45	**29.39**
MINSLK	36.53	30.31	**33.42**
EST	38.01	30.17	34.09
EFT	38.94	29.28	34.11
SPT	41.74	31.72	36.73
RANDOM	43.66	31.33	37.50
LPT	43.98	32.99	38.49

In the table 7 and 8, we have considered best four priority rules for S-SGS and P-SGS for analyzing scheduling the project by alternating between forward-backward scheduling. These performances based on all the combinations between the three best priority rules. The results are obtained from set of 30 activities with 4 type of resources, set of 60 activities with 4 types of resources, set of 120 activities with 4 types of resources. Table 7 and 8 shows the result of best four priority rules in forward and backward pass with respect to lower bound. The result shows both mean and standard deviation of best forward and backward heuristic priority rules. P-SGS with backward pass scheduling give the best result compare to S-SGS method.

Table 7.Best forward and backward heuristic priority rules from mean

Generation Scheme	**S-SGS**	**S-SGS**	**S-SGS**	**S-SGS**	**P-SGS**	**P-SGS**	**P-SGS**	**P-SGS**
Priority Rules	**LST**	**LFT**	**EST**	**MINSLK**	**LST**	**LFT**	**EST**	**MINSLK**
Set 30 activities 4 type of resources	29.14	28.97	32.56	37.57	25.53	25.70	30.53	33.15
	25.94	25.45	27.17	31.85	25.42	24.46	27.28	28.72
Set 60 activities 4 type of resources	28.54	28.45	35.13	38.63	26.75	28.42	32.09	30.95
	26.60	25.77	26.61	31.11	24.55	23.49	25.84	24.96

Set 120 activities 4 type of resources	40.04	41.63	55.24	49.61	38.70	38.65	51.43	45.50
	38.90	34.79	37.86	43.12	35.36	33.81	37.40	37.25
Average deviation	**31.53**	**30.84**	**35.76**	**38.65**	**29.39**	**29.09**	**34.10**	**33.42**

Table 8.Best forward and backward heuristic priority rules from standard deviation

Generation Scheme	**S-SGS**	**S-SGS**	**S-SGS**	**S-SGS**	**P-SGS**	**P-SGS**	**P-SGS**	**P-SGS**
Priority Rules	**LST**	**LFT**	**EST**	**MINSLK**	**LST**	**LFT**	**EST**	**MINSLK**
Set 30 activities 4 type of resources	23.79	23.67	24.54	26.39	22.37	22.10	22.96	25.83
	23.83	24.18	22.35	23.74	22.14	21.47	22.70	24.62
Set 60 activities 4 type of resources	30.20	31.59	33.00	37.29	28.09	29.30	30.37	31.87
	30.10	32.49	30.51	32.29	29.18	28.38	29.04	30.45
Set 120 activities 4 type of resources	37.77	39.82	37.73	40.91	34.92	35.81	36.17	39.21
	38.02	41.83	36.89	40.87	36.77	35.49	35.17	37.57
Average deviation	**30.62**	**32.26**	**30.84**	**33.58**	**28.91**	**28.76**	**29.40**	**31.59**

3.9 Parallel and Serial Method

In resource constrained project scheduling problem (RCPSP) the S-SGS requires more computational time than the P-SGS. In S-SGS each activity selected with decision set and tests all modes to find minimum feasible finish time. For a certain priority rule the mean CPU time used is more in S-SGS than P-SGS. We have observed from iterative backward and forward method, the parallel scheduling scheme is superior in comparison with serial scheduling method.

In order to test the significant of the parallel scheduling method and series scheduling method results, we performed 2-tailed Wilcoxon matched pair singed-rank test. This is well known non-parametric statistical test has the advantage over the t-test that we do not need to make any

assumptions about the distribution. Since none of the priority rules the percentage deviation from the optimal is normally distributed it is always better to test in non-parametric test.

The null hypothesis is $\mu_1 = \mu_2$ where μ_i is the average percentage increase above the critical path length obtained by the heuristic i. The alternate hypothesis is $\mu_1 > \mu_2$. Table 9 finds the values of Wilcoxon statistics W_{ij}. AsW_{ij}= - W_{ji} the table 9 gives positive Wilcoxon statistics. If the value of W_{ij} is greater than the critical value of Wilcoxon statistics for the chosen significance level, we should reject the null hypothesis and the alternative hypothesis is accepted i.e. μ_i is greater than μ_j. This shows that heuristic j outperformed heuristic i. From the table 9 at 1% significance level i.e. $P \leq 0.01$, four of the test priority rules (LFT, MINSLK, LPT,SPT, SIO) outperform the remaining three priority rules (EST, EFT, Random).

Table 9: 2-tailed Wilcoxon matched pairs signed rank test for priority rules

Priority rules	LFT	EST	MINSLK	EFT	LPT	RANDOM	SPT	SIO
LST	**0.434**	0.000	0.000	0.000	0.000	0.000	0.000	0.000
LFT		0.000	0.000	0.000	0.000	0.000	0.000	0.000
EST			**0.851**	0.000	0.000	0.000	0.000	0.000
MINSLK				0.001	0.000	0.000	0.000	0.000
EFT					**0.991**	0.000	0.000	0.000
LPT						0.000	0.000	0.000
RANDOM							**0.095**	0.000
SPT								**0.424**

1% Significant difference

Table 10 shows the performance of the priority rules tested when the parallel scheduling generation scheme (P-SGS) and serial scheduling generation scheme (S-SGS). The analysis carried out with 2-tailed Wilcoxon matched pair singed-rank test. Again the best priority rules are LST, LFT, EST and MINSLK with a percentage deviation with respect to the lower bound of 32.57%, 33.02%, 40.98% and 41.94 in S-SGS and 30.33%, 30.92%, 38.01% and 36.53% in P-SGS respectively. The mean ordering of S-SGS is similar to the ordering of P-SGS. The P-SGS has a

significantly better performance than S-SGS according to the 2 -tails Wilcoxon signed rank test with a significant level of p-value 0.014.

The remaining priority rules have a performance significantly worse than other priority rules. The better performance is maintained for all the priority rules evaluated expect for the rule SPT for P-SGS. The rules LST, LFT, EST and MINSLK are applied with the P-SGS obtain an average deviation is significantly lower than S-SGS. The reductions are 2.24%, 2.10%, 2.97% and 5.41% respectively. Figure1 show the parallel scheduling method performed better than parallel method with the priority rules.

Table 10.Parallel vs Serial Schedule Generation Scheme

Priority rules	Serial	Parallel	Different
LST	**32.57**	**30.33**	2.24
LFT	**33.02**	**30.92**	2.10
EST	**40.98**	**38.01**	2.97
MINSLK	**41.94**	**36.53**	5.41
EFT	46.05	38.94	7.11
LPT	46.51	43.98	2.53
RANDOM	53.75	43.66	10.09
SPT	56.76	41.74	15.02

The P-SGS has a significantly better performance than S-SGS according to the 2-tails Wilcoxon signed rank test with a significant level of 0.014.

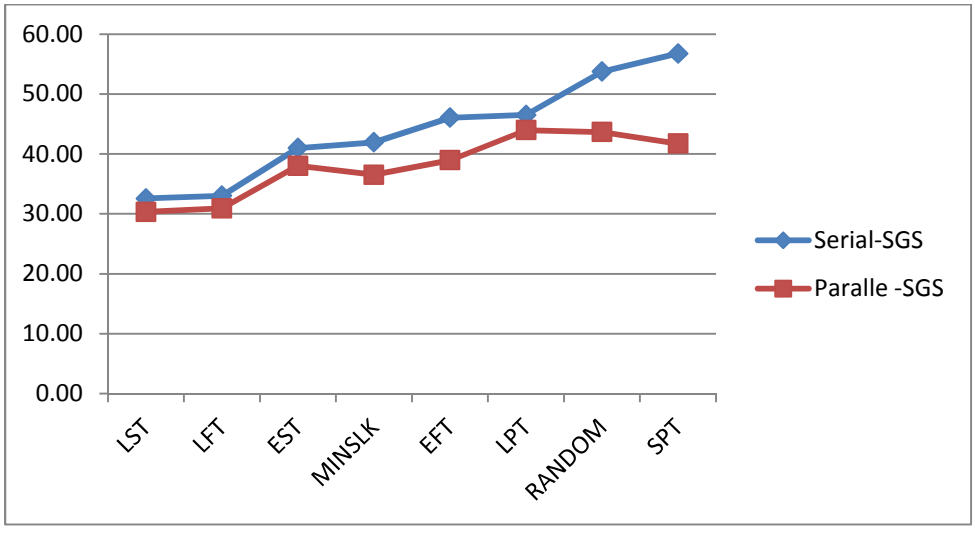


Fig. 1. Comparison between parallel and serial generation schedule with respect to priority rules

3.10 Effect of Problem Parameters on Performance of Priority Rules

Table 11 gives a brief summary of the project parameters values and the effect on the performance of priority rules over all 90 problems instances. Table 11 shows a single factor effect on the problem parameter on mean. The problem parameters i.e. network complexity (NC), resource factor (RF), resource strength (RS) are generated with all 90 instants. If only few precedence constraints in project i.e. NC = 1.5, the decision set gives more wrong scheduling decisions. If there are many precedence constraints, i.e. NC=2.1, reduce the number of schedule alternatives and the number of wrong answer is reduced. The problems with medium number of precedence constraints perform better as the number of precedence constraints and the number of scheduling alternatives is balanced. In NC, LST and STP priority rules perform better than all other priority rules.

The number of resource types required by each activity is measured by resource factor (RF).If there are very few resource types required by the activity, i.e. RF = 0.25, it is easy to take good scheduling decisions. Increasing RF to 0.75 or 1 means the problems are more complicated. The problems are averagely difficult if each activity are request RF = 0.5. In RS, LST and LFT priority rules perform better than all other priority rules.

Increasing resource strength (RS) means increasing resource strength that result for problems are easy to schedule. The priority rules LST and LFT perform better than other priority rules. Figure 2-4 shows the graphs of NC, RF and RS with the priority rules. From the figure parameter NC=1.5 and RF=0.5 and RS=0.70 LST and LFT performed better than all priority rules.

Table 11 Problem parameters performance on priority rules

		SPT	LPT	MINSLK	EST	LST	EFT	LFT	RANDOM
NC	**1.50**	71.90	71.42	76.94	74.78	77.92	76.86	81.18	82.34
	1.80	115.21	116.90	121.79	124.76	129.66	130.45	137.79	141.38
	2.10	126.09	126.55	136.09	142.73	142.55	146.09	149.27	157.82

		SPT	LPT	MINSLK	EST	LST	EFT	LFT	RANDOM
RF	**0.25**	98.16	92.45	88.71	88.45	83.23	91.35	83.26	95.68
	0.50	110.61	104.09	100.13	99.39	93.61	102.87	95.04	108.91
	0.75	115.06	104.28	102.17	102.83	95.11	106.72	94.67	112.83
	1.00	127.50	115.94	110.22	113.78	104.33	117.44	104.56	121.94
RS		**SPT**	**LPT**	**MINSLK**	**EST**	**LST**	**EFT**	**LFT**	**RANDOM**
	0.30	109.53	103.14	97.91	99.53	92.23	103.02	92.28	109.19
	0.50	121.52	110.66	108.45	107.00	100.41	109.90	101.31	117.28
	0.70	95.50	87.78	86.17	85.78	80.28	89.78	80.39	88.94

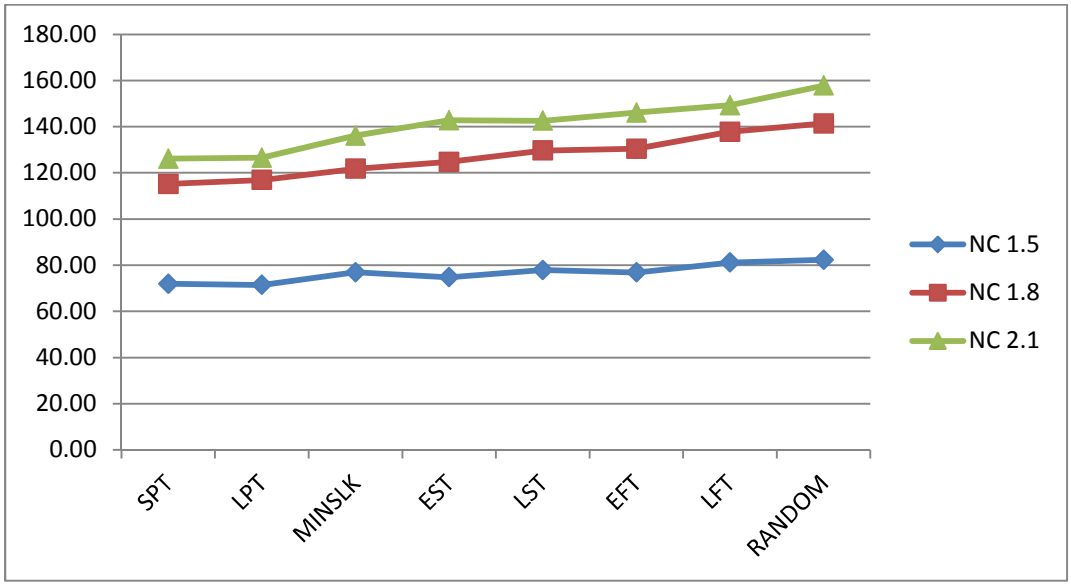


Fig.2. Network Complexity (NC) with Priority rules

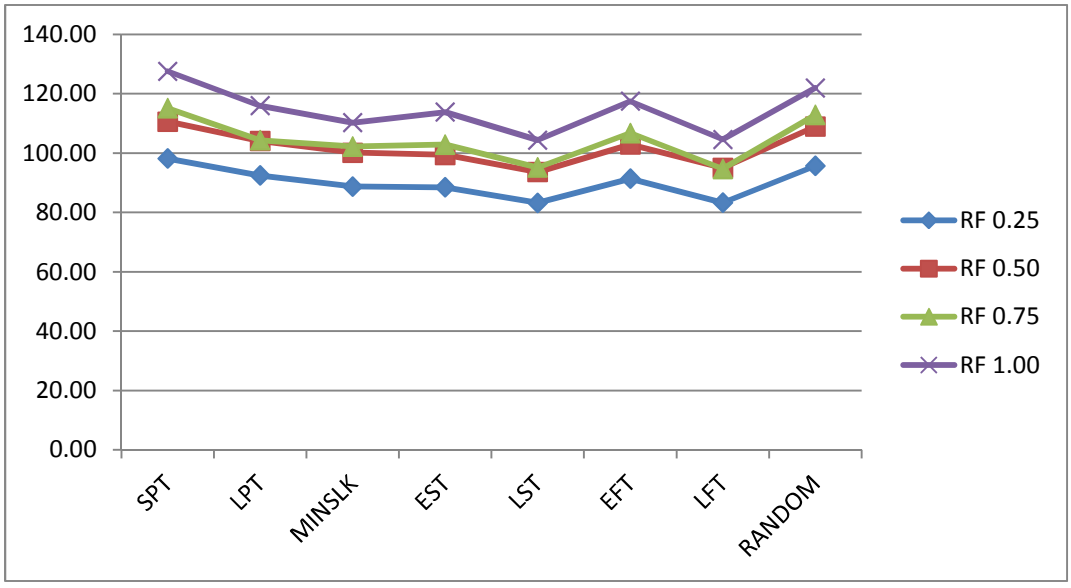


Fig. 3. Resource Factor (RF) with Priority rules

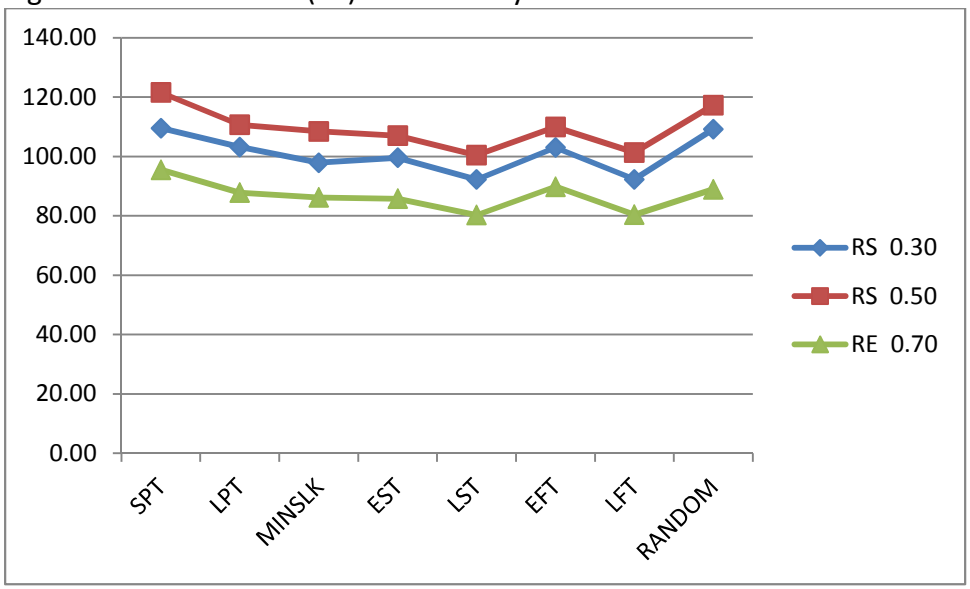


Fig. 4. Resource Strength (RS) with Priority rules

3.11 Robust Project Scheduling

The objective of resource constrained project scheduling problem (RCPSP) is to development of exact and sub-optimal procedures for the generation of workable baseline schedule assuming the deterministic environment. The project activities are scheduled subject to precedence constraints and resource constraints under the main objectives minimizing project duration. The baseline schedule serves very important functions for planning of external activities developed by Demeulemeester and Herroelen, (2002), Aytug et al. (2003), Mehta and Uzsoy (1998). The baseline schedules enable organizations to estimate the completion time of their projects and take corrective action when needed. It allows for scheduling and resource allocation decisions that the project complete in due date.

In project scheduling the common practice is to generate a baseline schedule before the start of the project. The deterministic project scheduling under resource constraints mainly focuses on the scheduled start times of the project activities, satisfies both the precedence constraints and the resource constraints and optimizes the scheduling objective most often the project duration.

3.12 Computation Results of Robust Scheduling

Each test problems are randomly selected in the PSPLIB library. Each of the 90 test problems is solved with each of the 33 heuristic tandems and the results are summarized in table 12. The STC heuristic generates a robust scheduling using activity duration distribution and activity inflexibility weights. A Monte Carlo simulation is performed to calculate the cumulative project duration distribution using the initial schedule as a starting point for project execution. Table12 shows an un-buffered schedule is generated from the lower bound optimal duration which has a service level of 95%. LST, LFT and EST shows the best heuristic serial single pass rules with average percentage increase over the critical path length is 53.21%, 53.65%, and 61.63% respectively with un-buffered schedule.

Comparing with baseline serial single pass schedule with respect to un-buffer serial single pass schedule the average percentage increase is 0.63%. The baseline completion due date will increase 0.63%, 0.62% and 0.51% in LST, LFT and EST respectively with service level 95% service. Table 13 shows the performance of priority rules parallel single pass rules. LST, LFT and EST shows the best heuristics with average percentage increase over the critical path length is 51.93%, 52.24%, and 57.67% respectively with un-buffered schedule. The baseline completion due date will increase 0.57%, 0.68% and 0.71% in LST, LFT and EST respectively with service level 95% service.

Table 12 Performance of priority rules serial single pass un-buffer

Priority rules	Mean	Deviation
LST	**53.21**	**31.81**
LFT	**53.65**	**32.79**
EST	**61.63**	**33.82**
MINSLK	62.48	35.33
EFT	65.82	33.72
LPT	66.52	34.37
RANDOM	74.46	36.20
SPT	75.54	37.95

Table 13 Performance of priority rules parallel single pass un-buffer

Priority rules	Mean	Deviation
LFT	**51.93**	**29.41**
LST	**52.24**	**29.17**
EFT	**57.67**	**29.70**
MINSLK	58.03	32.03
EST	58.55	31.11
SPT	61.85	33.29
RANDOM	64.75	33.94
LPT	65.06	34.32

Table 14 Performance of priority rules in serial backward and forward pass with un-buffer

Priority rules	Mean Forward	Mean Backward	Average BF
LST	**53.21**	**51.69**	**52.45**
LFT	**53.65**	**51.81**	**52.73**
EST	**61.63**	**51.59**	**56.61**
EFT	65.82	51.79	58.81
MINSLK	62.48	56.32	59.40
LPT	66.52	56.06	61.29
RANDOM	74.46	57.35	65.90
SPT	75.54	57.76	66.65

Table 15 Performance of priority rules in parallel backward and forward pass with un-buffer

Priority rules	Mean Forward	Mean Backward	Average BF
LFT	51.93	48.47	**50.20**
LST	52.24	49.84	**51.04**
EFT	57.67	50.34	**54.00**
EST	58.55	51.18	54.86
MINSLK	58.03	51.89	54.96
SPT	61.85	52.17	57.01
RANDOM	64.75	53.03	58.89
LPT	65.06	53.74	59.40

In this research, we have analyzed of the multi-pass heuristic based best forward-backward scheduling methods for un-buffer schedule. Table 14 and 15 shows average percentage deviation from the lower bound. Table 14 shows LST, LFT and EST perform better heuristic with average percentage of deviation 52.45%, 52.75% and 56.61% respectively for un-buffer schedule. Table 15 shows LFT, LST and EFT perform better heuristic with average percentage of deviation 50.20%, 51.04% and 54.00% respectively for un-buffer schedule. We have selected best four priority rules for serial scheduling generation Scheme (S-SGS) and parallel scheduling generation Scheme (P-SGS) for analyzing scheduling the project by alternating between forward-backward scheduling. The results are obtained from set of 30 activities with 4 type of resources, set of 60 activities with 4 types of resources, set of 120 activities with 4 types of resources.

Table 16shows the mean deviation with respect to the lower bound. The reports for different combination of SGS and priority rules are recorded in table 16. The lowest average deviation with respect to the lower bound with the combination of S-SGS is LST (52.45%) shows minimum from best three priority rules with un-buffer schedule. In addition, the lowest average deviation with respect to the lower bound with the combination of P-SGS is LFT (50.20%) shows minimum from best three priority rules with un-buffer schedule.

Table 16.Best forward and backward heuristic priority rules from mean for un-buffer

Generation Scheme	S-SGS	S-SGS	S-SGS	S-SGS	P-SGS	P-SGS	P-SGS	P-SGS
Priority Rules	LST	LFT	EST	EFT	LFT	LST	EFT	EST
Set 30 activities 4 type of resources	42.79	42.95	47.42	51.90	41.50	41.60	46.48	45.76
	41.33	41.07	42.42	41.09	39.64	40.83	40.08	41.72
Set 60 activities 4 type of resources	50.73	50.48	56.29	59.49	49.29	49.93	53.64	52.78
	48.53	47.78	48.59	48.80	44.76	46.42	48.38	47.40
Set 120 activities 4 type of resources	66.12	67.51	81.18	86.08	65.00	65.17	72.88	77.12
	37.39	66.57	63.77	65.48	61.00	62.27	62.57	64.41
Average deviation	**52.73**	**56.61**	**56.61**	**58.81**	**50.20**	**51.04**	**54.01**	**54.87**

Table 17 shows the analysis of 2-tailed Wilcoxon matched pair singed-rank test for P-SGS and S-SGS. The P-SGS has a significantly better performance than S-SGS according to the 2-tails Wilcoxon signed rank test with a significant level of p-value 0.014. In un-buffered scheduling the graph shows the parallel scheduling method performed better than serial method with the priority rules.

Table 17.Parallel vs Serial Schedule Generation Scheme for un-buffer

Priority Rule	Serial	Parallel	Diff.
LST	53.21	52.24	0.98
LPT	66.52	65.06	1.46
LFT	53.65	51.93	1.72
EST	61.63	58.55	3.08
MINSLK	62.48	58.03	4.45
EFT	65.82	57.67	8.16
RANDOM	74.46	64.75	9.71
SPT	75.54	61.85	13.69

The P-SGS has a significantly better performance than S-SGS according to the 2 -tails Wilcoxon signed rank test with a significant level of 0.014

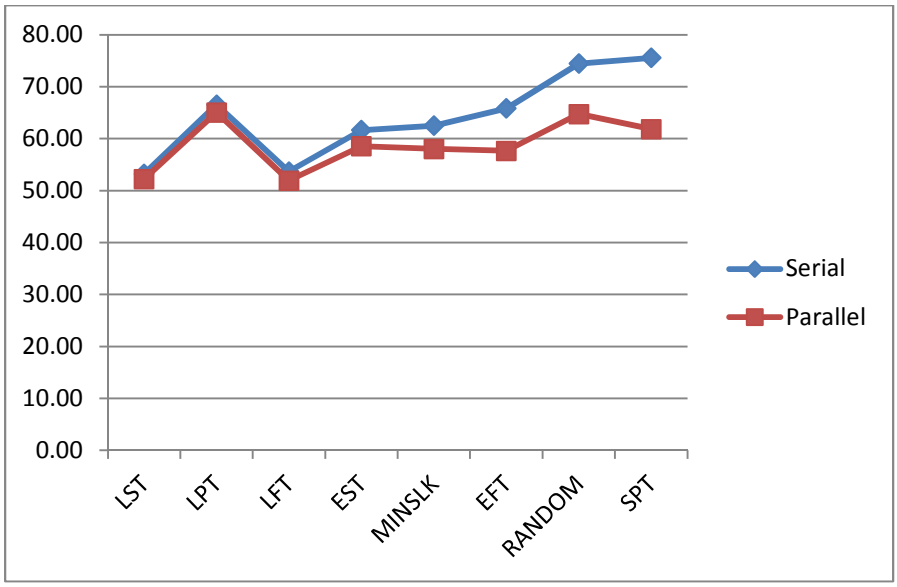


Fig. 5. Comparison between parallel and serial generation schedule with respect to priority rules for un-buffer

The problem parameters i.e. network complexity (NC), resource factor (RF), resource strength (RS) are generated with all 90 instants for un-buffer. In all problem parameters LST and LFT priority rules perform better than all other priority rules. Increasing Resource strength (RS) means increasing resource strength that result for problems are easy to schedule. The priority rules LST and LFT perform better than other priority rules. Figure 6-8 shows the graphs of NC, RF and RS with the priority rules for un-buffer. From the figures parameter NC = 1.5 and RF = 0.25 and RS = 0.50, LST and LFT performed better than all priority rules.

Table 18.Problem parameters performance on priority rules for un-buffer

Problem parameters		SPT	LPT	MINSLK	EST	LST	EFT	LFT	RANDOM
NC	**1.50**	118.30	114.20	109.30	102.34	102.96	111.76	103.74	119.14
	1.80	129.90	118.14	118.62	118.97	112.07	122.00	111.52	125.93
	2.10	135.45	122.64	122.55	122.55	127.27	127.27	116.27	128.64
		SPT	**LPT**	**MINSLK**	**EST**	**LST**	**EFT**	**LFT**	**RANDOM**
RF	**0.25**	95.84	92.58	87.97	89.74	84.00	91.84	83.39	95.26
	0.50	132.52	117.13	114.70	112.13	106.17	115.78	106.70	123.13
	0.75	150.44	138.44	132.89	134.61	125.33	137.67	126.72	145.39
	1.00	147.33	134.94	138.61	137.83	131.50	141.00	132.22	145.67
		SPT	**LPT**	**MINSLK**	**EST**	**LST**	**119.15**	**LFT**	**RANDOM**
RS	**0.20**	137.60	127.85	125.21	126.08	126.08	130.17	119.15	135.73
	0.30	116.29	110.08	107.71	106.42	106.42	107.83	101.25	114.58
	0.50	98.67	94.78	92.06	92.06	92.06	93.89	86.17	97.72

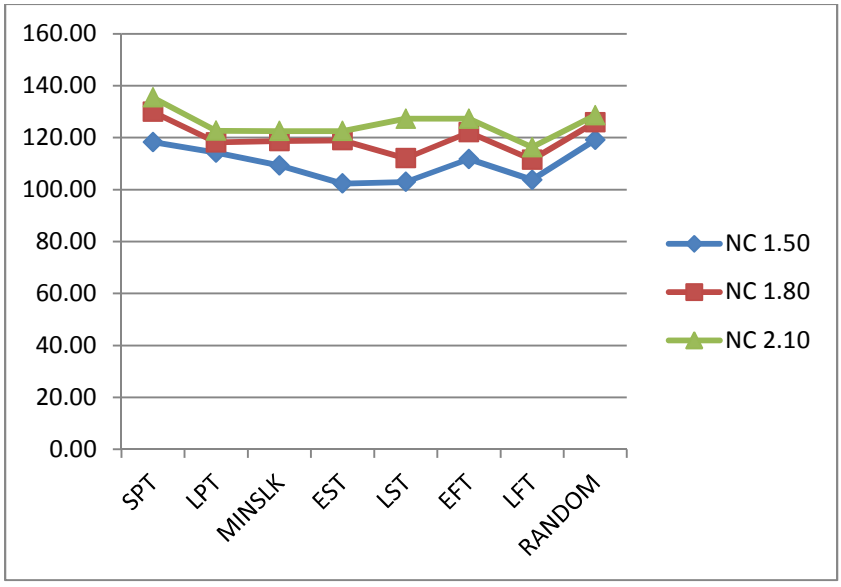


Fig. 6. Graph of Network Complexity with Priority rules for un-buffer

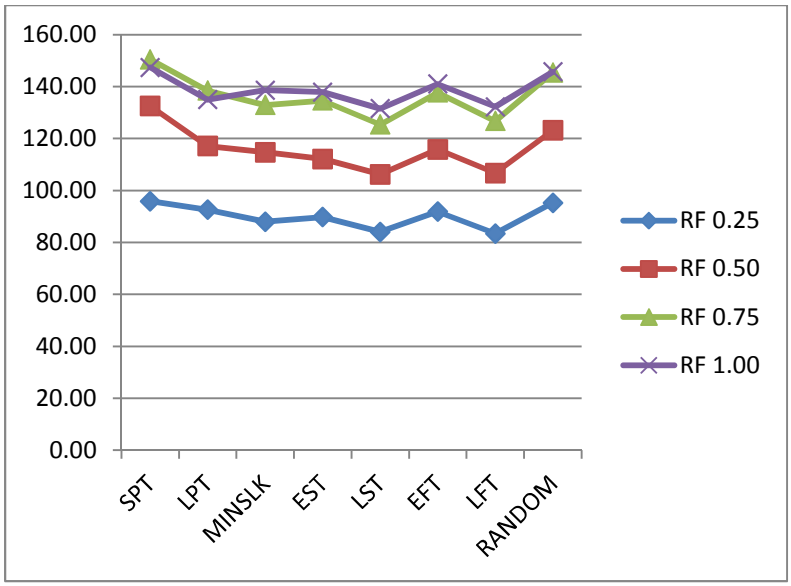


Fig. 7.Graph of Resource Factor with Priority rules for un-buffer

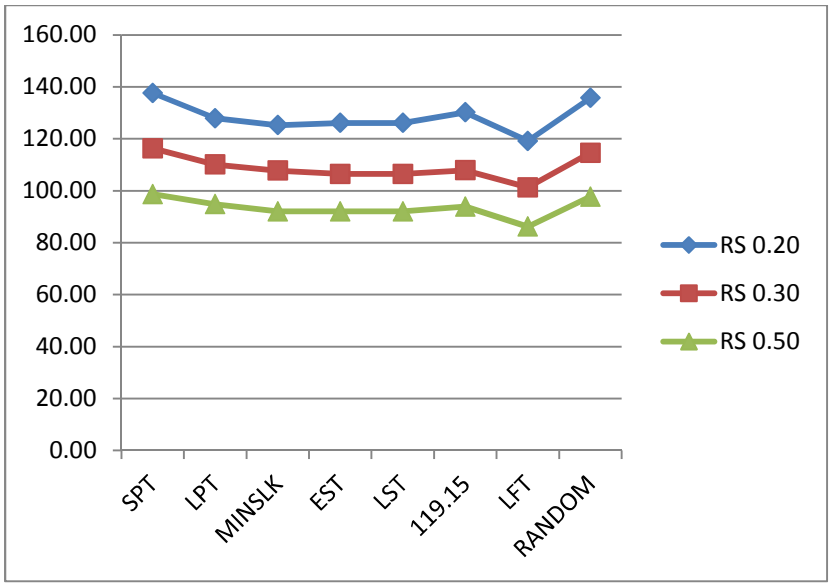


Fig. 8. Graph of Resource Strength with Priority rules for un-buffer

Chapter 4

4 Genetic Algorithms Approach for Resource Constrained Project Scheduling Problem

4.1 Introduction to Genetic Algorithms

Genetic algorithm (GA) is a stochastic heuristic search method whose mechanisms are based upon the principal of biological evolution processes discover by Charles Drawn in 1859. GA was first developed by J Holland (1975) that was based on the mechanics of natural selection in biological systems. It is an outstanding method to solve the complex optimization problems. The major benefit of this algorithm is that it provides a robust search in a complex space in compare with other optimization solution (Holland 1987). GA's are good for both local and global search of the search space because it operate on more than one solution space at once. Holland (1987) explained some of the characteristics of genetic algorithm compared to normal optimization search procedures are: (1) Smart search; (2) Progressive optimization; (3) Global optimal solution. (4) Black-box structure; (5) Good versatility; (6) Parallel type algorithm; (7) Intrinsic learning; (8) Stability.

Most of the genetic algorithms have three main operators 1) Selection 2) Crossover and 3) Mutation. The Genetic Algorithm is performed in the following steps:

Step 1: Population initialization

Step 2: Calculate the fitness value of each individual

Step 3: Selection

Step 4: Crossover

Step 5: Mutation

Step 6: Analyze the stop condition, if meet stop condition, go to step 7, else go to step 2

Step 7: Output the individual with best fitness value.

The Algorithm of GA:

```
Begin
  G=0
InitilizeP(g)
  Evaluate P(g)
Termination_condition =false
While termination_condination=false do
begin
    g=g+1
     Selects parents from P(g)
     Crossover
     Mutation
     Evaluate P(g+1)
end
end
```

4.2 Operators of Genetic Algorithm

Genetic Algorithms (GA) are the most popular technique in evolutionary computation research. In genetic algorithms, the representation used is a fixed length bit string. Each position in the string is assumed to represent a particular feature of an individual and the value stored in that position represents how that feature is expressed in the solution. The string is evaluated as a collection of structural features of a solution. The analogy may be drawn directly to genes in biological organisms. Each gene represents an entity that is structurally independent of other genes.

4.2.1 Initial Population

Population generation can be executed randomly or by seeding. Seeding is generating the initial population by high quality solution obtained from another heuristic technique. Seeding can help GA finding better solutions rather more quickly than it can do from a random start. However, there is a possible disadvantage in that the chance of premature convergence may be increased (Revees 1992 and Kapsalis et al. 1993).

4.2.2 Selection

The selection phase plays an important role in driving the search towards better individuals and in maintaining a high genotypic in the population. The chromosome that process better fitness value will have higher probability of appearing in the next generation. The roulette wheel strategy is used for selection for individual which ensures that the probability of selection of the fittest candidate will be the maximum.

4.2.3 Crossover

The most important genetic recombination operator is called crossover. Crossover is the necessary operation for the genetic reproduction. New genes are reproduced from randomly selected genes. Couples, namely the parents, are determined by randomly generated numbers and new two genes are reproduced from parents by crossover operation. The location of the crossover is also determined by generating a random number. After the crossover new two gene combinations are generated by the existing gene combination of the population.

A commonly used method, called one-point crossover, multi-point crossover which exchange portions of their representation. Eshelman et al. (1989), worked on multipoint crossover that examined the biasing effect of traditional one-point crossover and considered a range of alternatives. Eshelman (1989) concluded that simple crossover has considerable positional bias and the bias may be against the production of good solutions. In addition to this, crossover operator is analyzed in detail by Faily (1991). In multiple point type of crossover operations, the

gene is divided into equal intervals, which is same with the crossover point number. For each interval simple crossover operator is executed.

4.2.4 Mutation

Mutation prevents domination of a certain gene which has high probability of survival. Crossover can produce good fit genes from existing genes, but it cannot generate a new gene for a specific portion which does not exist in the population. Therefore, mutation operator has significant importance as it can produce new gene combinations, which is not generated at the initialization of the population.

4.3 Genetic Algorithm for Resource Constrained Project Scheduling Problem

4.3.1 Background Study

The resource constrained project scheduling problems (RCPSP) is one of the most challenging and grate particle applications in product development, product planning, construction planning and scheduling etc. For over the decades, RCPSP has received attention of researchers for solving these problems with various exact methods i.e. mathematical programming, dynamic programming, zero-one programming and branch and bound method using mathematical models but the disadvantage is that it could not solve the bigger and more complicated problems in practice.

4.3.2 Problem Definition and Formulation

RCPSP consists on executing a group of activities limited by constrains. Each activity in a project has a corresponding duration and also needs certain amount of resources such as labor or materials to execute. The precedence relationship is force to some activities to begin after the finalization of others. In addition, processing every activity requires a predefined amount of resources, which are available in limited quantities in every time unit. The aim of the limited resources the shortest duration is to make the daily resources requirement no more than daily supply, to make full use of limited resources and make the total duration as short as possible.

The assumptions of these problems are that the availability of resources is constrained to some maximum value that the project has to complete using the given resources. The RCPSP is characterized by a factorial growth in the amount of computation required to consider all possible solution as the problem size increases.

The objective function is formulated for exploring a solution to the RCPSP. The objective function are minimizing project duration, minimizing total project cost, maximizing net present value etc. Minimizing project duration is used in this study as an objective function RCPSP.

The objective of resource constrained project scheduling is to minimize the duration time of a project without exceeding the resource limits. Resource constrained allocation is one type of sequencing problem. Using GA to solve, a string of population represents a possible sequence of activities and each characteristic in the string stand for an activity ID (or name) activity in the lower position has a higher priority for allocation of resources.

RCPSP is describe as a project having a set of n+2 activities that need to processed j = 1, 2, ...n, n + 1, n + 2. The activity 1 and n + 2 are consider dummy activity marked start and end activity. These activities are not considered in resource allocation and having a duration of 0 time unit. The structure of the project is represented by an activity-on-node network G = (V, E) in which V denotes the set of vertices (node) representing activities and E is the set of edges (arcs) representing the finish-start precedence relationships with zero time-lag. Activities are related with two types of constraints. First, precedence relations are finish-start (FS) type that is j activity may just be started when i activity is concluded. For each activity j there is a predecessor activity P_j are all less than j, that is, for P_i € P_j, it needs to meet the condition i < j. A second type of constraint is related to resources. Activities consume resources that are provided in limited quantities. Suppose the completion of the activity i, j € (1, 2 … n+2) needs kth resources.

There are a total of k renewable resources available during whole duration of project. Resources are in the set k = 1, 2 ... m. Resource is limited with its capacity R_k in each time period. The duration of the activity denoted as d_j and the amount of resource k used by the activity j in each time unit is denoted as $r_{t,k}$ units of k, (k=1,2..m*)*resource type during each period of duration d_j

(there is only one way of processing the activities). The activities cannot be interrupted once they are started.

The d_j, r_{jk}, R_k parameters are supposed as known, determined and integer; also, $d_1=d_n= 0$ and $r_{1,k}= r_{nk} = 0$ for every k(k = 1,2…m). The RCPSP problem instance is solved if the starting times (or ending times) for each activity satisfying the precedence and resources constraints, in a way that the duration of the project is minimized.

The objective of RCPSP is to find minimum project duration for each all resources constraints are met. The optimization model for the minimum project duration with resource allocation as follows:

Minimize $f(i) = \text{Max}(t_i + d_i \mid i = 1, 2...n)$ (1)

Subject to

$t_j - t_i \geq d_i$ for all S_j (2)

$\sum r_{t,k} \leq b_k$, A(t), (k= 1,2,…m), $t \geq 0$ (3)

$f(i) \geq 0$ (4)

Where f(i) is the fitness function of an individual, which means the finish time of activity i; t_i is the starting date of activities i, d_i is the duration of activity i, S_i is the set of successors of activity i, $r_{t,k}$ is the resource demand of k^{th} of activity i at the duration of t, b_k is the resource limit of the k^{th} resource, m is the total number of resources type and n is the total numbers of activities.

Equation (1) expresses the computation of the project duration, which is the objective function of the RCPSP. Equation (2) takes the precedence relations into consideration between each pair of activities (i ,j), where i immediately precedes j. This constraint is the difference between the starting times of successor activities and those of predecessor activities should be equal to or greater than the duration of the connecting activity. Equation (3) limits the total resource usage within each period to the available amount. This constraint express that the sum of all available resources allocated to different activities on the same day should not exceed the total number of units available on that day.

Finally, Equation (4) states that the fitness value, which is the finish time of an activity I of an individual schedule, is always equal to or greater than zero. RCPSP, however, became a solvable problem in a polynomial time if resource constraints are eliminated. The starting times for each activity are calculated as follows:

$s_1 = 0$; and $s_i = \max (s_i + d_i)$, i =1,2 ... n.

This method generates a sequence denominated ES (earliest start) sequence, where every activity is sequenced as early as it can be, taking into account the precedence relationships. ES_i is the starting time for i activity in ES sequence and EF to $ES_i + d_i$ that is the earliest ending time for j activity.

4.4 Proposed Genetic Algorithm for Resource Constrained Project Scheduling

GAs is a robust general purpose search program based on the mechanism of natural selection and natural genetics. In this research, selection presents an elitist GA to search for an optimal solution to the resource constrained project scheduling of large project networks by implementing serial scheduling generation scheme as a decoding procedure. The elitist GA is proposed using following operators: definition phase, evaluation, elitist selection, one-point crossover, uniform mutation, fitness calculation and termination. GAs typically works with the collection or population solutions rather than with a single solution. The initial population of possible solutions to the RCPSP is prepared at the very beginning of the operation. GAs, potential solution to the problem of RCPSP is represented as a population of chromosomes and each chromosomes stands for a possible solution. Each individual solution is represented by a single string like entity called a chromosome. A chromosome typically consists of a number of genes.

These offspring chromosomes are created by merging two parent chromosomes using a crossover operator or modifying a chromosome using a mutation operator. A new generation is formed by selecting some of the parents and offspring according to the fitness values and rejecting the others so as to keep the population size constant. During each generation, the chromosomes are evaluated on their performances with respect to the fitness functions (objective functions). Fitter chromosomes have higher survival probabilities. This process is continued for a large number of offspring generations in which the population is evolving (better solution replace unfit solution) until a terminating criteria is met. The final chromosomes hopefully represent the optimal or near-optimal solution to the problem.

4.5 Permutation based Encoding and Decoding

4.5.1 Definition Phase

The proposed algorithm is a permutation based encoding (Hartmann 1998). This is called an activity list representation. A schedule representation is a representation of a priority structure among the activities. A solution for the RCPSP is represented in a chromosome that represents an activity sequence for the problem. A chromosome is also called an individual that is given by an activity sequence. Each gene in a chromosome stands for an activity number. An activity has a lower priority than all preceding activities in the sequence and a higher priority than all succeeding activities.

Each individual in the population is represented by an array of activities in the project. In figure 9, it has shown the activity list representation for a project with N activities. Activity j will be the ith activity chosen to be scheduled. It will be scheduled in its earliest feasible start time. When activity j is chosen to be scheduled, all its predecessors, which will appear in some position 1, 2 . . . i — 1 will have already been scheduled. In this way, the related schedule will always be a feasible schedule. In the definition phase, we have defined the resource required, the duration to finish each individual activity, predecessor relationship of each activity.

We developed a random number generator to generate precedence feasible permutation in Visual C++ program coding. The random number generator creates an initial population for

possible solutions to the RCPSP. For example, an individual (3,1,4,2,5,6,8,9,7,10) has 10 activities. It does not determine the fitness value, but it defines a possible starting and finishing time of an activity. The fitness value for the project is obtained from the maximum value out of all fitness values of every activity to be scheduled in a project. The project duration is obtained by comparing the finish time of the last activity and the fitness value of the activity. Table 19 shows example of a project definition phase.

1	….	…	…	i	…	…	N
	….	…	…	j	…	…	

Fig 9: Activity list representation

Definition Phase											
Define activity no	1	2	3	4	5	6	7	8	9	10	11
Define predecessor activity	0	0	0	1	4	2	2	6	3	5	7
Define resource required	3	6	4	2	4	2	4	3	4	5	2
Define duration of activity	4	6	2	8	4	10	16	8	6	6	10

Table 19: Definition Phase of the project

4.5.2 Serial Scheduling Generation Scheme

The serial schedule generation scheme was developed by Kelly (1963). In RCPSP, a serial schedule generation scheme is the mechanism to compute the fitness value as well as the start and finish times of the activities in an individual. The purpose of applying the schedule generation scheme to the individuals in a population is to obtain schedules that showed the resource profile and the project duration. Fig 10 shows project duration with the resource profile. It consists of the n stages, which is the same as the number of activities to be scheduled. A set of activities to be scheduled can be classified as two disjointed activity sets: scheduled set and decision set. In scheduling set S_n the activities that are already scheduled are belong to the partial schedule. The decision set D_n contains the unscheduled activities with every predecessor being in the scheduled set.

For example of an individual of 11 activities which is randomly generated. Table 20 shows the randomly generated activities. As per the randomly selected activities in each individual, the first

activity selected based on the sequence. Activity 2 is selected in stage 1 first because it is first position of all activities in the individual, I. It has zero predecessor relationship so it can start from zero. Activity 2 is scheduled at time 0 for the duration 6 days as it required resource is not exceeding the availability of resources. Fig. 10 shows the serial scheduling generation scheme for all stages.

I =

2	6	3	1	4	7	9	5	11	10

Table 20: Randomly generated activities

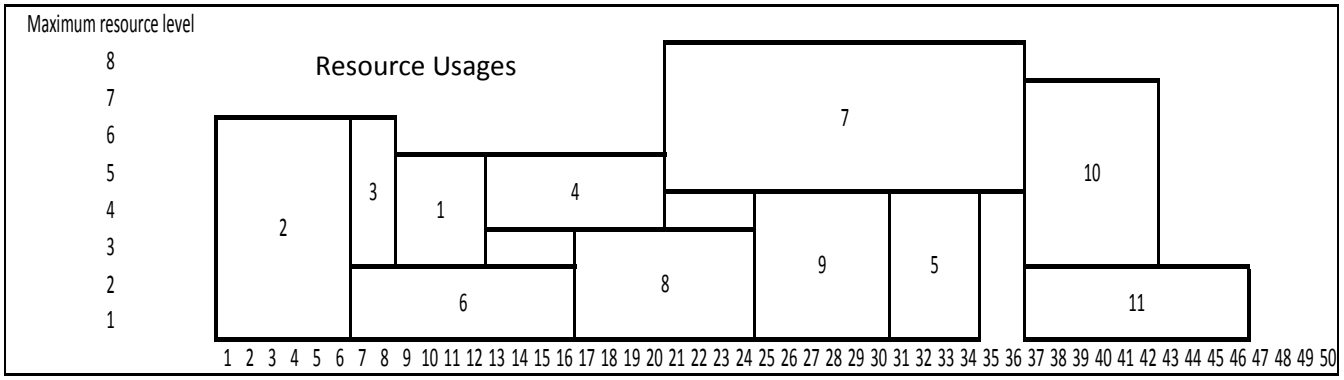
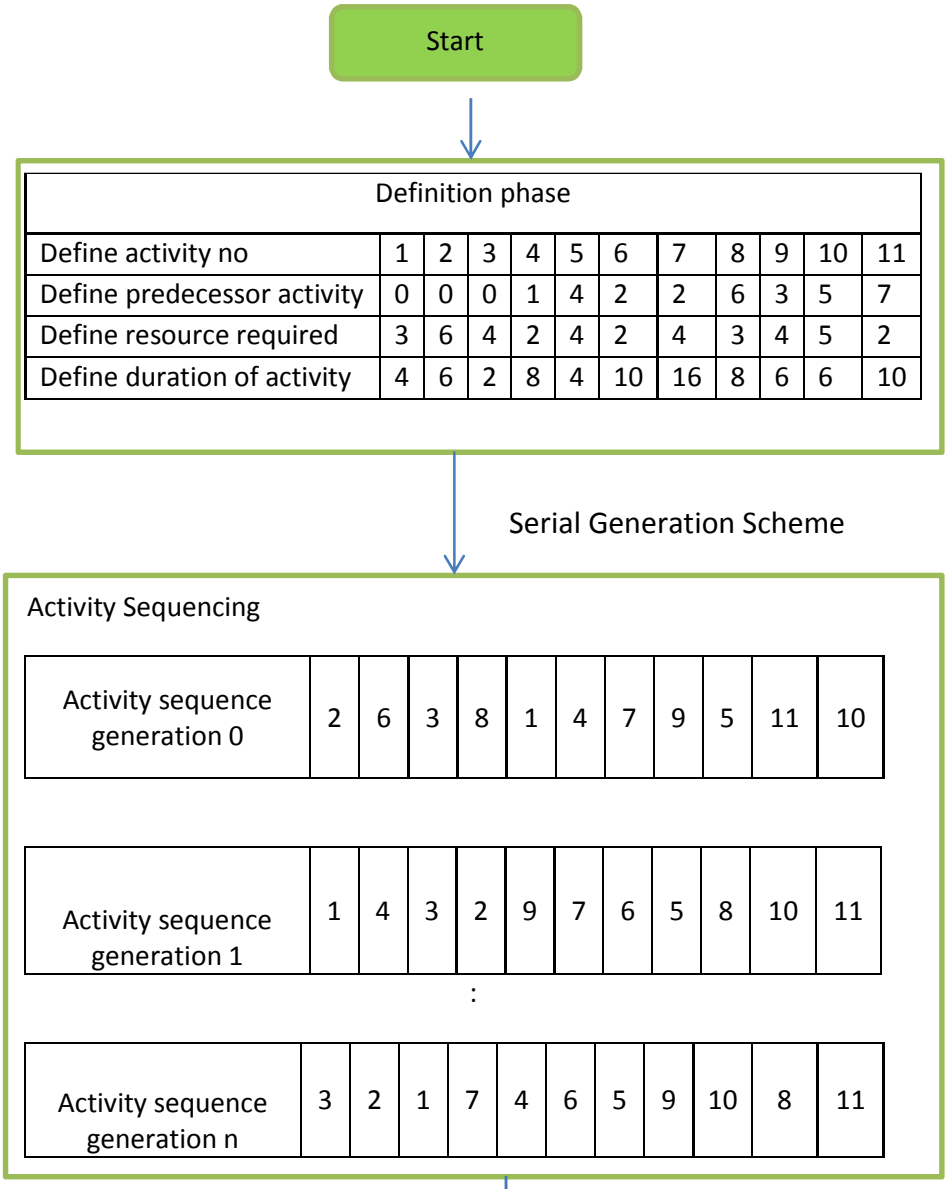
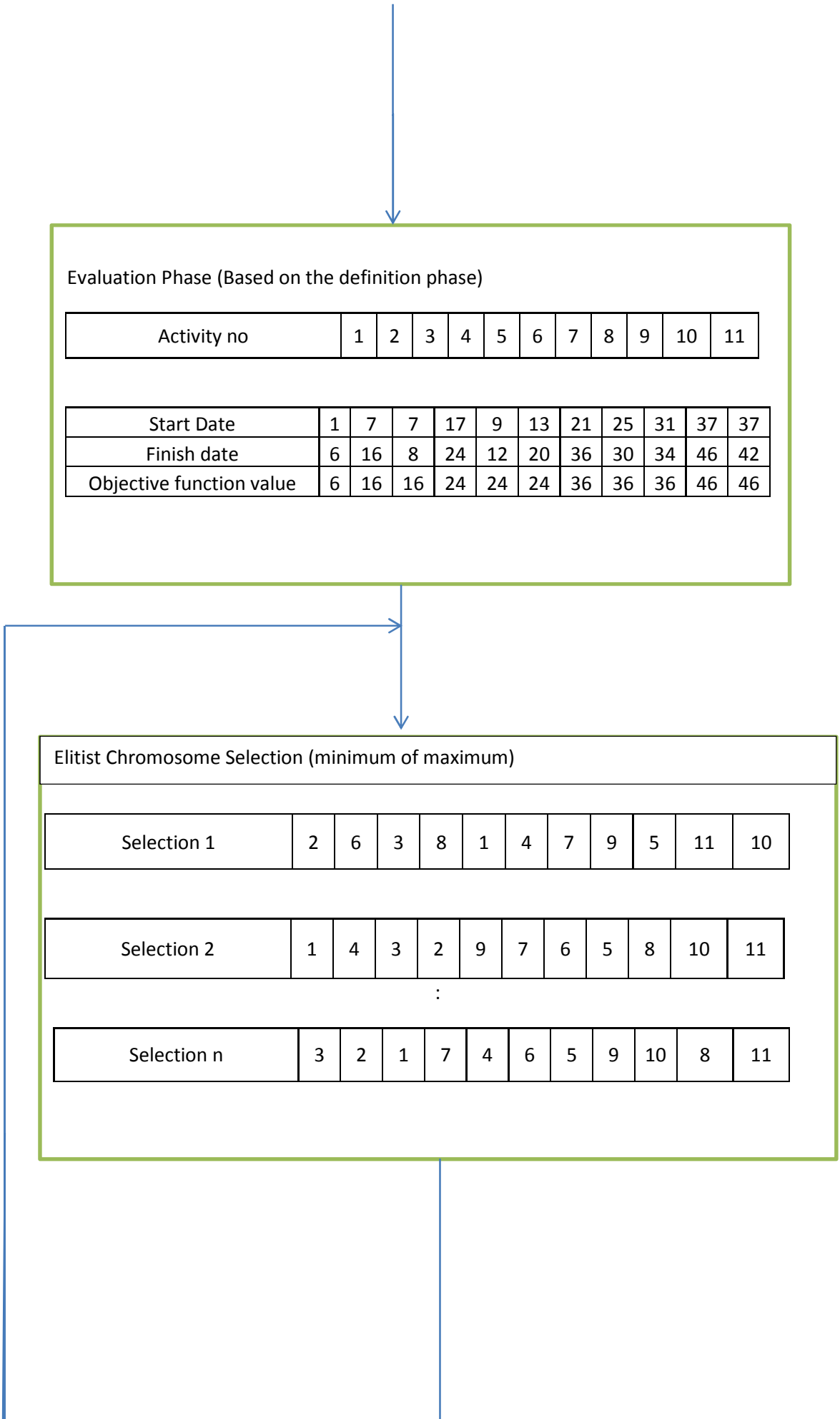


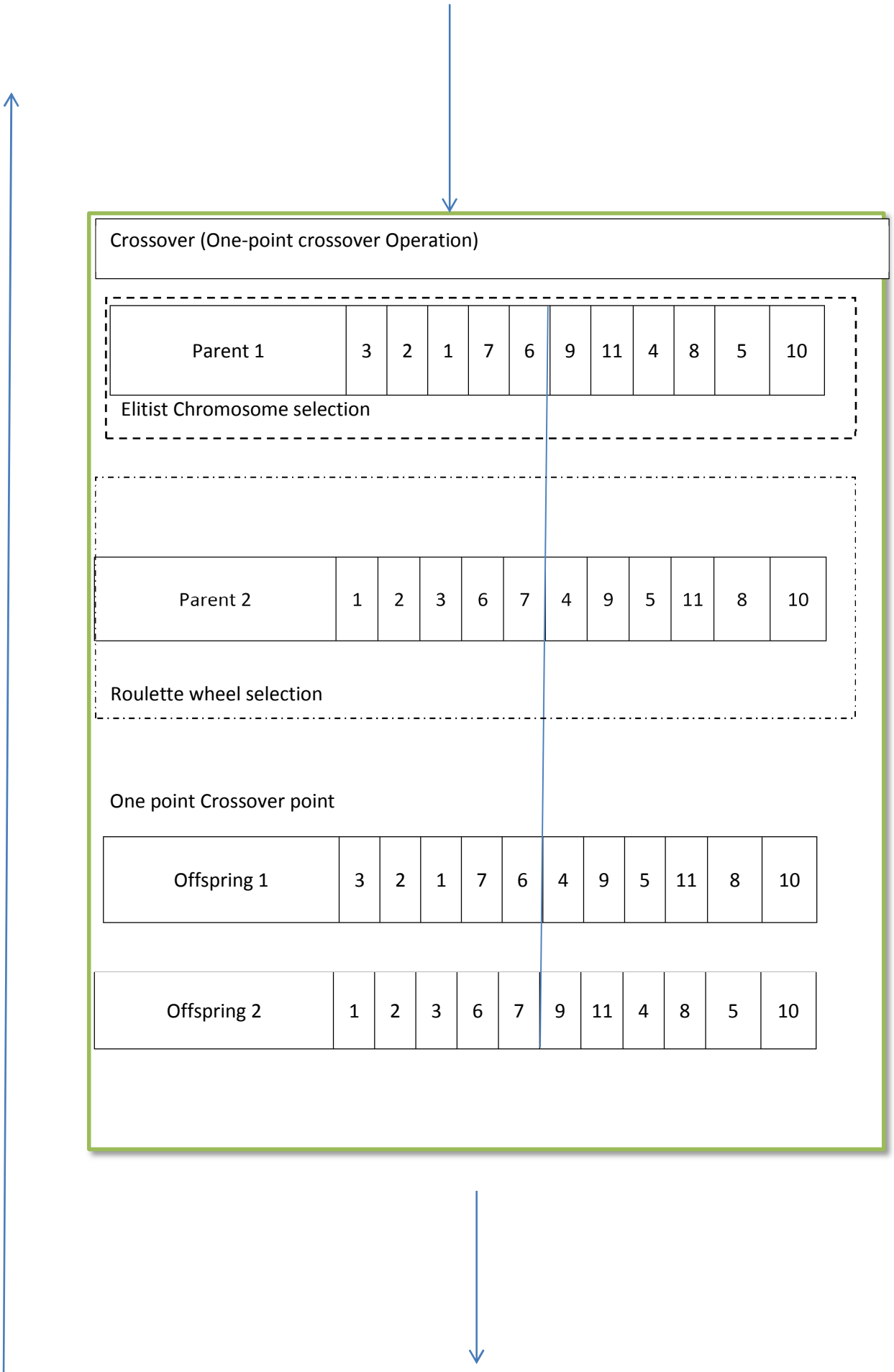
Figure 10: Serial schedule generation scheme

The activity 2 required 6 resources hence the resource left is 2 as the maximum available resource is 8. In stage 2, activity 6 is next selected as per the sequence list of individual. Activity 2 is the immediate predecessor of activity 6. The activity 2 is completed hence the activity 6 carry forward for the activity schedule. If the predecessor activity does not completed in the scheduling process then it is selected next activity in sequence list of individual and continue till it found predecessor activity exit in the sequence. In stage2, the resource requirement for the activity 6 is 2 units and duration to complete is 10 days. The resource required for the activity 6 is less than available resource. So the schedule is between time 7 and time16. The resource left after the activity 6 schedules is 6 units in this span of time frame.

In stage 3, activity 3 is selected as per the sequence list of individual. Activity 3 can be scheduled at the earliest possible time because it has no precedence relationship with any other activities. Activity 3 cannot be scheduled at time 0 due to the limitation of available resources. Activity 3 requires 4 resources. Therefore, activity 3 is scheduled at time 7 as its resource requirements 4 do not exceed its resource availability throughout the period being scheduled. The activity 3 schedules for 2 days from time 7. The serial scheme continues in the same way up to the last activity in the individual is scheduled.


Start
Definition phase
Define activity no 1 2 3 4 5 6 7 8 9 10 11
Define predecessor activity 0 0 0 1 4 2 2 6 3 5 7
Define resource required 3 6 4 2 4 2 4 3 4 5 2
Define duration of activity 4 6 2 8 4 10 16 8 6 6 10
Serial Generation Scheme
Activity Sequencing
Activity sequence generation 0 2 6 3 8 1 4 7 9 5 11 10
Activity sequence generation 1 1 4 3 2 9 7 6 5 8 10 11
:
Activity sequence generation n 3 2 1 7 4 6 5 9 10 8 11


Evaluation Phase (Based on the definition phase)
Activity no 1 2 3 4 5 6 7 8 9 10 11
Start Date 1 7 7 17 9 13 21 25 31 37 37
Finish date 6 16 8 24 12 20 36 30 34 46 42
Objective function value 6 16 16 24 24 24 36 36 36 46 46
Elitist Chromosome Selection (minimum of maximum)
Selection 1 2 6 3 8 1 4 7 9 5 11 10
Selection 2 1 4 3 2 9 7 6 5 8 10 11
:
Selection n 3 2 1 7 4 6 5 9 10 8 11


Crossover (One-point crossover Operation)
Parent 1
3 2 1 7 6 9 11 4 8 5 10
Elitist Chromosome selection
Parent 2
1 2 3 6 7 4 9 5 11 8 10
Roulette wheel selection
One point Crossover point
Offspring 1
3 2 1 7 6 4 9 5 11 8 10
Offspring 2
1 2 3 6 7 9 11 4 8 5 10

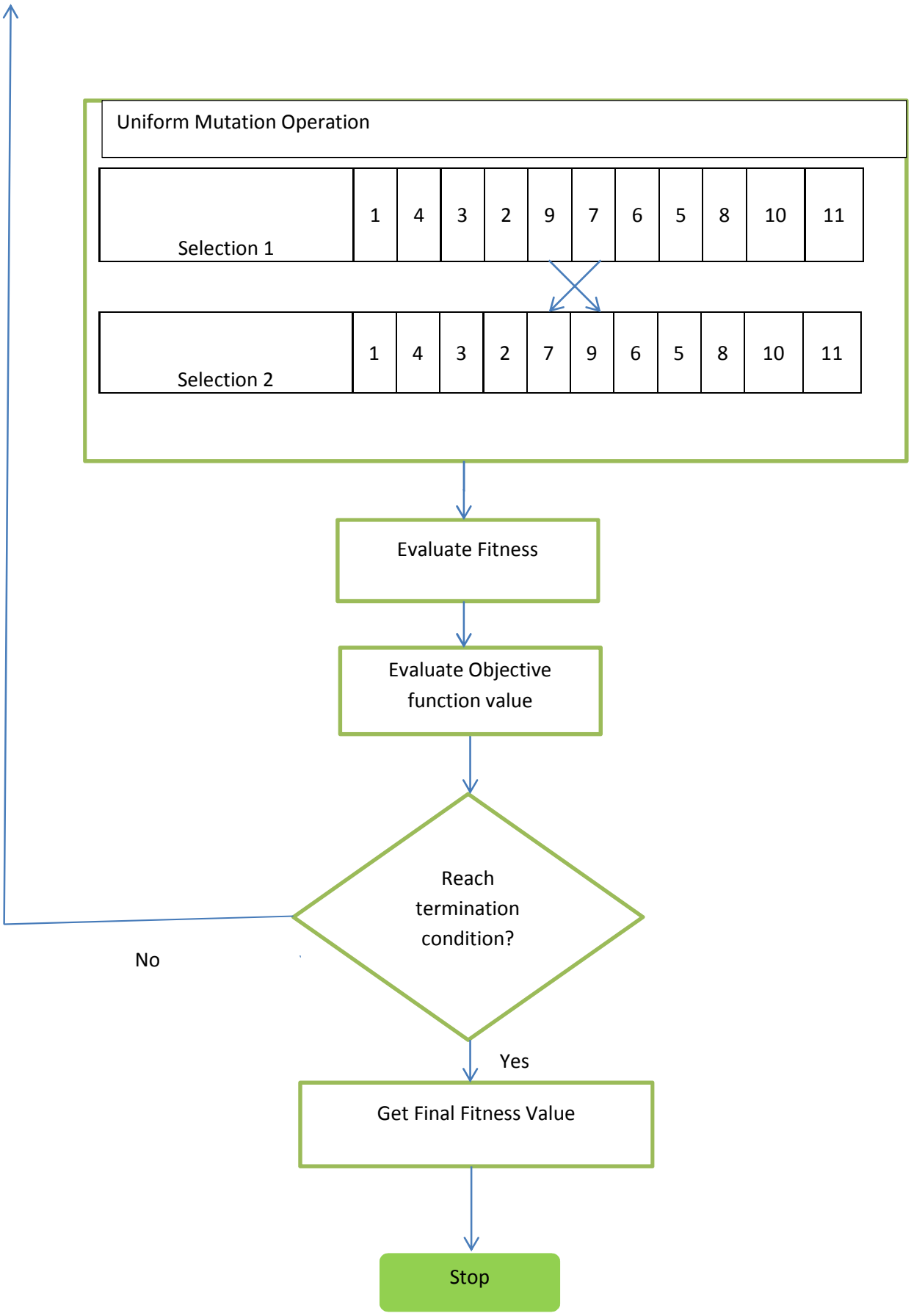

Uniform Mutation Operation
Selection 1
1 4 3 2 9 7 6 5 8 10 11
Selection 2
1 4 3 2 7 9 6 5 8 10 11
Evaluate Fitness
Evaluate Objective function value
Reach termination condition?
No
Yes
Get Final Fitness Value
Stop

Fig 11: Operation of the proposed algorithm

4.5.3 Elitism Selection

Selection is the procedure by which better than average solutions are determined for recombination. The selection criterion is used to choose parent chromosomes for crossover operations. The proposed algorithm is select the elitist chromosomes with minimum of maximum strategy that find the best chromosome to the new population, whereas the rest of the population remains as normal. It then selects parent chromosomes using roulette wheel selection method.

4.5.4 One Point Crossover

The crossover operation corresponds of concept of mating. It combines pieces of information coming from different individuals in a population. It is the principal mechanism by which GA arranges for good schemas present on different chromosomes to a single individual. The variation will be less, if the starting times for schemata are fixed for the genotype because at time 0 the number of activities can be start is limited with the precedence relationships. For example, schema of the genotypes (1, 3, 2, *, *, *, *, *, *, *, *, *) or (1,2, 3, *, *, *, *, *, *, *, *, *) will be likely to schedule at the same starting time if they do not violate the resource constraints.

In RCPSP, mostly two different types of crossover operators are used for cross over operations. The union crossover operator 3, UX3,Leu and Yang (1999) and the one point crossover Hartmann (1998), are identified as good methods for the permutation-based encoding for the solutions to the RCPSP. In our algorithm we have explained about one point crossover operation. The one point crossover operator is capable of preserving schemata in a more effective manner because it keeps the first half of both parents intact and random. The section one point crossover computed as per the objective function value.

In this proposed algorithm, we have selected one point cross over operator for permutation based encoding for RCPSP. First, the parent 1 individual is selected by elite strategy that coping the best chromosome based on the objective function. Second, parent 2 individual is selected based on the roulette wheel selection method.

4.5.5 Proposed One Point Crossover algorithm

The proposed crossover point selection algorithm will enable the program to select more random offspring. This randomization process will increase the search space of the fitness function. By this cross over algorithm, the increased randomization enable the program to search the fitness value in more wide space and this will increase the convergence between number of generation and standard deviation with larger space.

To find out the crossover point in both the individuals the activity number simultaneously has to keep adding. In every step the cumulative sum has to be compare. If the value found to be equal then the respective sequence position has been stored. The process carry forward till the penultimate activity in the sequence and in each case the respective sequence number has been noted. The cross over point is the closest to the center of the sequence which is having the probability 0.5. It has been considered to select the sequence point closest to the center point of sequence to balance the randomization and the predetermined sequence structure determined based on the definition and predecessor relation. In fig 12, we will consider the first case for more randomization of activities and both is equal distance from the center.

One point cross over algorithm

parent 1											
activity no in sequence	1	4	3	2	9	7	6	5	8	10	11
cumulative sum of activity no	1	5	8	10	19	26	32	37	45	55	66
parent 2											
activity no in sequence	2	6	3	8	1	4	7	9	5	11	10
cumulative sum of activity no	2	8	11	19	20	24	31	40	45	56	66
parent 3											
activity no in sequence	1	3	4	2	9	6	7	11	8	5	10
cumulative sum of activity no	1	4	8	10	19	25	32	43	51	56	66

Considering 1st and 3rd sequence											
Before crossover											
parent 1	1	4	3	2	9	7	6	5	8	10	11
parent 3	1	3	4	2	9	6	7	11	8	5	10

After crossover											
case 1											
offspring 1	1	4	3	2	9	6	7	11	8	5	10
offspring 3	1	3	4	2	9	7	6	5	8	10	11

case 2											
offspring 1	1	4	3	2	9	7	6	11	8	5	10
offspring 3	1	3	4	2	9	6	7	5	8	10	11

Fig 12. One point crossover operation

4.5.6 Uniform Mutation Operation

The uniform mutation is changing two neighboring genes without violating the precedence relationship. Mutation is performed at the gene level but not every gene should be mutated. In this algorithm, for each individual generation, the operator generates a real random number and then swaps an activity after pivot point with activity if a random number is equal to or less than mutation probability. A mutation on an individual does not necessarily change the related schedule because interchanging two activities that have the same start time in the activity sequence.

4.5.7 Termination Conditions

The proposed algorithm terminates to produce the best solution that contains the starting and finishing time of all the activities in the individual, resource fitness values, the project duration, sequencing date. Table 21 shows the calculation of fitness values. The termination conditions include the number of generations which can be set up at the beginning of the algorithm.

4.5.8 Proposed algorithm for Computing the Fitness Value of Each Individual

1. Define the project activity number, predecessor activity, resource requirement and activity duration.
2. Get the random sequence of activity number by random number generator operator in C++ program coding.
3. Select each activity one by one as per the sequence generated randomly.
4. For each selected activity get the predecessor activity, if the predecessor activity is zero it implies that the activity can start at a specific time.
5. Based on the predecessor activity number and the binary function it has to check whether the predecessor activity is actually exist or not. If the binary operator is zero the predecessor activity does not exist and in this case the selection will carried forward for next activity to repeat the same, else if it is one then that activity sequence has been selected.
6. The start time has been calculated, it is the time greater than the finish date of the predecessor activity.
7. Base on the resource availability and available time span (which has to be fit to the activity duration defined in definition phase) the start time and finish time has been calculated.
8. The finish time has been calculated for each activity.
9. The fitness value is the maximum value of each activity finish date.
10. The available resource based on the different activity start and finish interval has been calculated using C++ program coding. The sequenced date is the finish date of some activity and duration is the time span from its previous duration, the available resource is

the resource available for the next sequenced activity to perform based on the random sequencing.

11. On activating each activity this available resource, time span and sequenced date adjusted based on the start and finishing time of each activity.
12. The final fitness value is the maximum fitness value of each population of sequenced activity string.
13. The fitness value will be calculated based on each population fitness value.

The minimum value of each population fitness after cross over and mutation will be the objective function value.

4.5.9 Experimental Results and Analysis

The permutation based elitist GA has been programmed using Visual C++ programming language coding and tested on Pentium 4 CPU GHz processor under the Windows 7 operating system. The computational experiments are explained to demonstrate the performance of the proposed algorithm. The parameters for the proposed algorithm include population size, crossover probability, and mutation probability for global search.

The population size, crossover, and mutation rates are taken, respectively, 200, 0.5, and 0.05 for the solution. Increasing populations size generally result in better solution, but it required more computer memory. The algorithm terminated with the number of generation of 200. Table 21 shows computational result of randomly generated individual and the schedule. It is tested using a single resource constraint with one point crossover and uniform mutation. The result shows that the fitness value is 46 days. When no crossover is occurred, the computation result cannot obtain the optimal value 46 days. So the one point crossover operation worked well in this algorithm. We have taken two example of randomly generated individual. After running GA the fitness value obtained is 46 days in both the cases. Fig 13 and 14 are shows the resource allocation with one type of resource constrained and the fitness value of individual.

Table 21: Profile of the elite individual 1

Sequencing to calculate objective function											
Activity sequence generated by random no generator	2	6	3	8	1	4	7	9	5	11	10
Binary operator	1	1	1	1	1	1	1	1	1	1	1
Start Date	1	7	7	17	9	13	21	25	31	37	37
Finish date	6	16	8	24	12	20	36	30	34	46	42
Objective function value	6	16	16	24	24	24	36	36	36	46	46
Sequenced date	6	8	12	16	20	24	30	34	36	42	46
Time span	6	2	4	4	4	4	6	4	2	6	4
Available resource	2	2	3	4	3	1	0	0	4	1	6

Fig.13: Calculation of objective function of each population string of individual 1

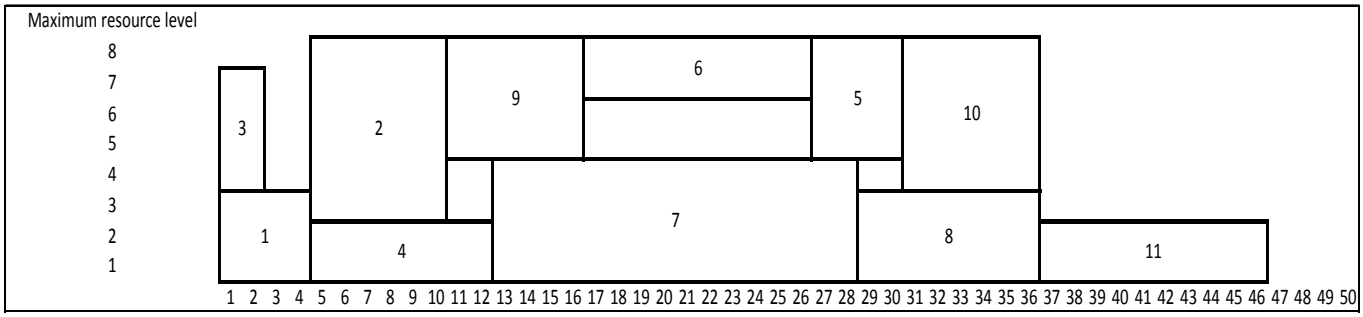


Table 22: Profile of the elite individual 2

Sequencing face to calculate objective function											
Activity sequence generated by random no generator	2	6	3	8	1	4	7	9	5	11	10
Binary operator	1	1	1	1	1	1	1	1	1	1	1
Start Date	1	7	7	17	9	13	21	25	31	37	37
Finish date	6	16	8	24	12	20	36	30	34	46	42
Objective function value	6	16	16	24	24	24	36	36	36	46	46
Sequenced date	6	8	12	16	20	24	30	34	36	42	46
Time span	6	2	4	4	4	4	6	4	2	6	4
Available resource	2	2	3	4	3	1	0	0	4	1	6

Fig. 14: Calculation of objective function of each population string of individual 2

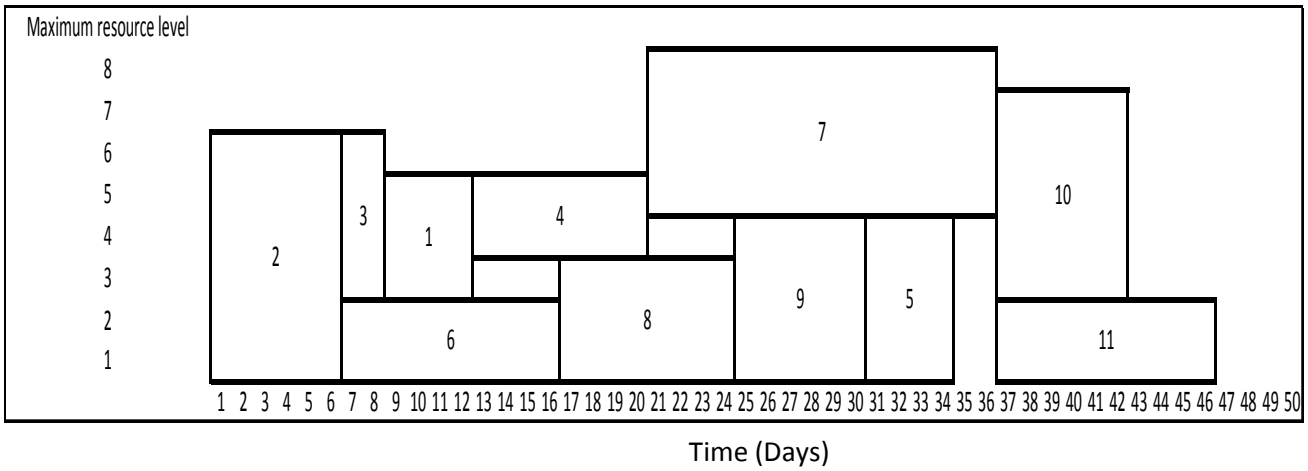


Figure13 and 14 shows the comparisons results optimal and lower bound solutions.

4.5.10 Conclusions

RCPSP is normally characterized by project duration length as objective function and features of resource constrained and permutation condition. A scheduling generation scheme is most important for the heuristic scheduling process. This research addressed a permutation-based elitist genetic algorithm for searching for the optimal solutions to the large-sized multiple resource constrained project scheduling problems. The proposed algorithm will produce a suboptimal and/or optimal solution to the large-sized multiple RCPSP.

The proposed algorithm is searched competitive solution as compare to the heuristic method. The proposed algorithm is a good combination of activity-list representation and elitist strategy to search optimal and/or near optimal solution to the RCPSP. The performances of the GA operators used in this research shows that the algorithm demonstrated its performance and accuracy through the comparisons with optimal and lower bound solutions. Hence the proposed genetic algorithm is a powerful tool for scheduling single and multiple resource constrained projects.

Chapter 5

5 Crashing Stochastic PERT Networks with Resource Constrained Project Scheduling Problem

5.1 Introduction

Project Management is the process of planning, controlling and monitoring the activities associated with a large complex project. The success of any project is very much dependent upon the quality of planning, scheduling and controlling of the various phases of the project. Planning and controlling projects is of paramount importance in the success of any project, because it constitutes a major part in the project management life cycle. Critical Path Method (CPM) and Program Evaluation and Review Technique (PERT) are very common and widely adopted project management tools. CPM does not provide a measure of uncertainty and it is developed under the condition of normal working on the project duration time. PERT is an aid to management in expending and controlling the resources to meet the scheduling completion date of the projects which involve high degree of uncertainty.

5.2 Crashing

Crashing refers to compression project activities which is the decreasing total project schedule duration. Crashing project duration take place after analyzing all possibility activities are crashed, resource availability and least addition cost. The objective of crashing a network is to determine the trade-off between optimal project duration and minimum project cost. Each phase of the crashing project duration design to consume resources hence cost is associated with it. The cost will vary according to the amount of time consumed by the design phase. An optimum minimum cost project schedule implies lowest possible cost and the utilization of resources associated time for project management.

5.3 Literature Review

PERT problem was developed by Naval Engineers of USA Navy in charge of Polaris sub-marine missile project in 1958. Euro working group on project management and scheduling was

established to assess the state of art and to emulate progress in theory and practice. CPM and PERT about project management have been inadequate in research papers and text books. Steve and Dessouky (1977) described a procedure for solving the project time/cost trade-off problem of reducing project duration at a minimum cost. The solution to the time and cost problem is achieved by locating a minimal cut in a flow network derived from the original project network. Dobin (1984) also determined the most critical paths in PERT networks.

Johnson and Schon (1990) used simulation to compare three rules for crashing stochastic networks. He also made use of criticality indices. Badiru (1991)developed simulation program for project management called STARC. STARC allows the user to calculate the probability of completing the project by a specified deadline. Kelfer and Verdini (1993) gives the expediting projects in PERT with stochastic time estimation. A detailed survey of resource constrained project scheduling has been given by Ozdamar and Ulusay (1995). Pulat and Horn (1996) described a project network with a set of tasks to be completed according to some precedence relationship; the objective is to determine efficient project schedules for a range of project realization times and resource cost per time unit for each resource. Cho and Yum (1997) measured an uncertainty importance of activities in PERT network. Cho and Yam(1997) developed method for “Uncertain Important Measure”. It has been intended to explore some of the problems for crashing stochastic networks. Sevkina and Hulya (1998) tried to analyze on project management using primal-dual relationship. Rehab and Carr (1989) described the typical approach that construction planners take in performing time-Cost Trade-off (TCT).

5.4 Problem Definition

In the project management generally there is a specific date for the project completion. In order to complete the project in less than the normal time, the normal duration of the project must be reduced to the desired duration. The method of reducing the project duration by shortening time of one or more activities at a cost is called crashing. Network crashing developed in CPM for reducing duration of the project with the additional amount of money invested. It selects the lowest cost slope activities which will reduce the time in the critical path(s). The idea of crashing

a project with resource constrained was not developed in PERT where probabilistic time was used. We have focused on the problem of crashing the project duration with resource constraint using PERT network.

This objective of this research has two fold. First, develop a model to constructing a relationship between project duration and the cost i.e. time-cost tradeoff with resource constraint environment. Resources are available only in limited quantities and the resource demands of concurrent activities may not be satisfied. Second, network crashing is developed in CPM for reducing duration of the project with the additional amount of money invested. It selects the lowest cost slope activities which will reduce the time in the critical path(s). The idea of crashing a project was not developed in PERT where probabilistic time was used. This research focuses on the problem of crashing the project duration using PERT network with resource constrained with additional amount of money invested in each activity.

5.5 Mathematical Formulation

PERT calculates the expected value of activity duration as an average of the three time estimations. It makes the assumption that the optimistic, most likely time and pessimistic activity times a, m and b respectively. So the average or expected time t_e of an activity is determined from beta distribution.

A project consists of a set of activities which have to be processed. The activities are interrelated by two kinds of constraints. First, precedence constraints each activity can not to be started before all its immediate predecessor activities in the set have been finished. Second, performing the activities requires renewable resources with limited capacities. The activity durations are independent continuous random variables; preemption is not allowed. The objective of the problem is to crash the PERT network with renewable resources for finding minimum makespan of the project.

PERT activity has three times estimation: optimistic (a), most likely (m), pessimistic (b). For commuting the expected duration of each activity the three times estimation assumes follow the beta distribution. β- distribution is a probability distribution with density function $K(t-a)^{\alpha}$ $(b-t)^{\beta}$. The expected time (mean time) t_e is located at one third of the distance between most likely time and mid-range. So, the expected time t_e of the activity will be

$$t_e = (a+4m+b)/6 \qquad (1)$$

and standard deviation $$\sigma = \{(b-a)/6\}^2 \qquad (2)$$

The formulas (1) and (2) are supposed to be valid for the variable t_e that has a variety of distribution shapes.

The project completion time can be reduced by reducing (crashing) the normal completion time of critical activities. The reducing normal time of compulsion will increase the cost of the project. However, decision maker always look for trade-off for the total cost of a project and the time required completing. It is easy to compute crash the activities and the completion time for the project with unlimited resource availability.

In this research, we have developed a model for crashing PERT network with resource constrained condition. The resources are defined for each activity. The mathematical model is

Min $\sum c_i x_j$ i =1,2…n

Subject to

$x_j - x_i \geq t_{ij}$

$\sum r_{t,k} \leq b_k$ j € A(t) k=1,2…m

x_i= earliest even time of node i

x_j= earliest event time of node j

t_{ij}= time of activity i→j

c_i= crashing cost

$r_{t,k}$ is the resource demand of k^{th} of activity i at the duration of t

b_k is the resource limit of the k^{th} resource

m is the total number of resources type

n is the total numbers of activities

We have considered crashing the project activity with resource availability and resource requirement. The experimental model is defined with a number of activities with resource constrained environment. The project has 40 activities with three times estimation. The resource availability and requirement of each activity is given in the table 25.

5.6 Experimental Model

The developed algorithm is evaluated using a large number of networks. A construction company wants to make a new commercial market complex and it has been contracted a time period of 300 days to complete which is a normal time period required to complete the project. The budget of the project is $7.25 million and some additional amount of money is available.

But the management wants to finish before two week so as selling without any delay. Also, the management wants to spend some extra for the early completion of market complex. As the activities of this project are uncertain time duration and resource constrained, the project manager have construct a PERT network model for planning and scheduling of the project with availability and the requirement of resources. But it is essential to complete the project around two weeks before the duration, so as to complete within their allotted time duration and satisfy the management.

Table 23 shows the three time estimates of the activities of the project: optimistic (a), most probable (m) and pessimistic times (b). In each activity requirement and availability resources are defined in table 25. Figure 15 shows the network model using activity on arrow diagram. As the activities of this project are uncertain time duration, the project manager has constructed a

PERT network model for planning and scheduling of the project. This is an experimental model for decrease the time duration with resource constrained in PERT network.

Table 23: Three time estimation of activity duration of the project

Activity No.	**a, m, b**	**Activity No.**	**a, m, b**	**Activity No.**	**a, m, b**
A1	16,18,25	A15	9,16,19	A29	16,18,23
A2	10,12,18	A16	12,14,16	A30	12,14,16
A3	12,14,18	A17	8,10,13	A31	3,7,19
A4	9,12,16	A18	10,15,18	A32	5,11,14
A5	5,8,18	A19	10,12,15	A33	6,10,16
A6	13,14,18	A20	7,13,16	A34	7,12,15
A7	12,13,15	A21	4,6,9	A35	18,20,25
A8	8,11,16	A22	9,12,16	A36	10,14,16
A9	10,12,17	A23	21,22,25	A37	9,12,16
A10	9,10,12	A24	8,12,14	A38	11,13,16
A11	8,13,17	A25	9,12,17	A39	16,19,24
A12	6,12,28	A26	15,17,20	A40	12,18,21
A13	12,15,17	A27	11,16,18		
A14	5,7,11	A28	8,13,17		

Table 24: Crashing duration and cost

Node	Normal time	Crash time	Cost/unit crash ($)
1	0	0	0
2	18.83	4.71	600
3	12.67	3.17	450
4	14.33	3.58	500
5	12.17	3.04	400
6	13.17	3.29	300
7	12.50	3.13	300
8	14.50	3.63	350
9	12.84	3.21	370
10	14.83	3.71	400
11	10.17	2.54	250
12	13.67	3.42	350
13	14.67	3.67	400
14	15.33	3.83	550
15	12.17	3.04	350
16	12.50	3.13	200
17	22.33	5.58	500
18	12.00	3.00	150
19	12.33	3.08	200
20	17.17	4.24	250
21	12.83	3.21	100
22	18.50	4.63	400
23	10.33	2.58	100
24	14.00	3.50	170
25	11.67	2.92	150
26	20.50	5.13	370
27	19.33	4.83	400
28	17.50	4.38	350

Table 25: Resource availability and resource required

Nodes	**Resource required**	**Resource availability**
1	0	0
2	3	5
3	3	4
4	3	4
5	3	4
6	3	4
7	2	3
8	3	4
9	3	5
10	3	4
11	2	3
12	2	4
13	3	4
14	4	5
15	3	5
16	3	4
17	3	5
18	3	4
19	2	3
20	4	5
21	2	3
22	3	5
23	2	3
24	3	4
25	3	4
26	3	4
27	3	4
28	3	4

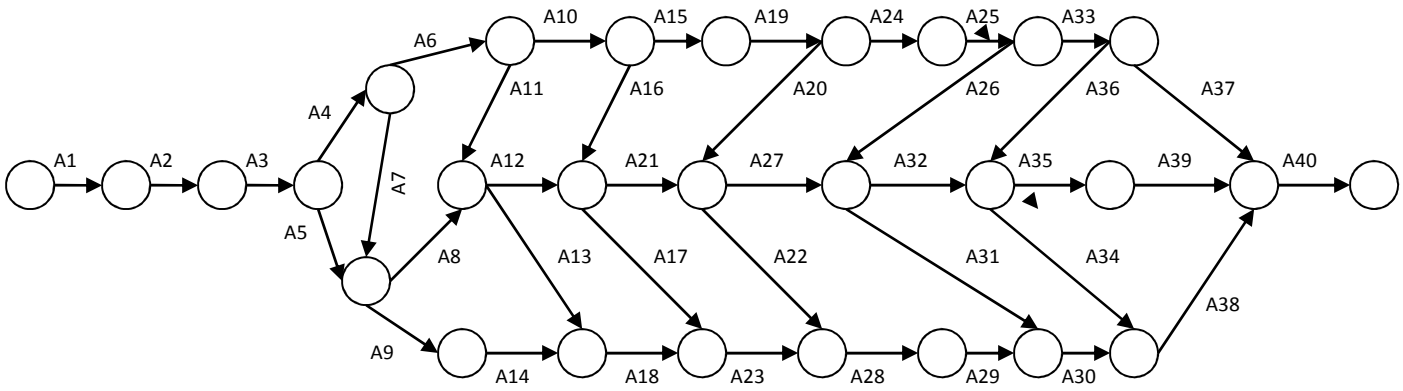
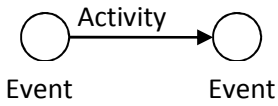


Fig. 15 Activity on Arrow (AOA) network



Notation:

i - Start event ' i' for an activity

j - End event 'j' for an activity

ES - Earliest Start time

EF - Earliest Finish time

LS - Latest Start time

LF - Latest Finish time

x_i- earliest even time of node i

x_j- earliest event time of node j

resave- resource availabilityfor activity

resreq- resource required for activity

xrand- Random number

5.7 Proposed Algorithms for Crashing PERT Networks with Resource Constrained

(1) Input data for the project i.e. activity duration, cost, resource availability, resource requirement.

(2) Calculate earliest time estimates for all the activities.

$ES_j = \max(ES_i + \text{duration}_i)$ for all x_j, $j \geq 1$

(3) Calculate latest time estimates for all the activities.

$LS_i = \min(LS_j - \text{duration})$ for all x_i, $i \leq x_j$ where $i = n$

(4) Calculation of slack: $\text{slack}_i = LS_i – ES_j$ for all x_n

(5) Initialization ES(1)=0

(6) Calculation of early finish time:

$EF_j \geq EF_i + \text{duration}_j + (\text{duration}_j/\text{resave}_j) * \text{resf}_j - \text{crash}_j$

(7) Resource reserved: $\text{resf}_j = \text{resreq}_j - \text{resava}_j$ for all x_j

(8) Generating the random number for unit cost for unit resource.

$\text{xrand}_j = \text{rand}(\text{resf}_j)$ for all x_j

(9) Is early finish time is less than equal to due date then calculate minimum cost

Min =sum (ccost*crash*xrandom_j*resf_j)

Otherwise go to step 6

5.8 Computational Result

The critical path of the project and the starting and finishing times of the individual activities, ES, EF, LS, LF are calculated. The project activities are schedule in resource constrained conditions. The project is crashed according to cost slope. The program is looking for the resources availability and cheeks each activity's resource requirement. If the required resource is more than the availability, the program delayed the activity to the following day. The process continued until the calculation of the finish time of the last activity.

The lower bound of the project (critical path method) of the project without resource constrained condition is calculated. The project completion time without resource constrain is

251 days. The resource requirements for the project activities depend on the availability daily total resources. The resources availability and requirement for the activities of the project are given by in table 25. The crashing duration and the cost of the activities of the project are given in the table 24.

The developed algorithm is coded in LINGO 11.0 software for crashing network with activity duration and resource constrained condition. Additional amount of money is assigned for each activity for the project. The crashing time of each activity is uniformly taken as 25% of the original duration. The algorithm runs 60 iterations with 0.01 CPU time. The optimal duration obtained for crashing the network with resource constrained is 282 days with addition amount of money $ 58,884.92.

The relative percentage of deviation from the original critical path (CPM) duration is calculated. The optimal solution of the model is evaluated in PERT network without resource constrained is 251 days. Relative deviation for the project is calculated by:

RD(%) = (RCD-d)/d x 100

Where RD = relative deviation of resource constraint duration from CPM duration in percentage;

RCD= resource constraint duration

d = original CPM duration without resource constraint

In this model relative deviation from CPM is calculated by using above formula. The relative deviation of resource constraint duration from original CPM is 17%. The objective of this model is to compute the minimum project completion duration with resource constrained condition. The optimal solution shows that duration increase 17% more with resource constrained condition from original CPM duration.

5.9 Summary and Conclusion

The concept of crashing in PERT networks can be applied on several networks by investing additional amount of money. The objective of this constructed model is to increase the realizing the last node by minimizing pessimistic time, which led to decrease in project duration. This model is tested by investing addition amount of money with limited resources. The completing time is reducing to 282 days. Moreover, the schedule time decreased from 294.15 to 282 days for the additional amount of money $ 58,884.92 with the availability of resources. So it is not possible for the company to complete the project before 282 days with the availability of resources. Therefore, we have to spend more additional amount so as to complete the project exactly before one week. This algorithm will help project planners to investigate crash duration and cost in resource constrained condition in PERT network.

Chapter 6

6 Conclusions and Future Research

6.1 Summary of Research Conclusions

The complexity of the decision-making processes clearly challenges managers as well as researchers to focus on models and tools which help them to take decisions in complex environment. The integer programming, goal programming, dynamic programming, and simulation model could be powerful tool for decision-making process. These tools cannot search optimal solution to large and medium size of problems. Obtaining an optimal schedule for the resource constrained project scheduling problem is NP-hard. This means that the problem cannot solved in a computationally optimization process. The amount of time needed to solve such an NP-hard scheduling problem increases with the problem size and it is intolerable for scheduling large and complex problem. Thus heuristic procedures have received much attention and have been shown to be computationally very efficient.

The performance of heuristics scheduling rules is significantly affected by network structure, performance criteria and the degree of resource tightness. None of the heuristic rules can always produce the best solution for all problems. Computing the heuristic rules for single pass method with four type of resource constrained scheduling LST, LFT and EST rules is most effective. Two other rules SPT and SIO are not performance well in compare to other rules. These three rules are LST, LFT and EST generally better than the other rules. The result may have limited to the type of problems having hundred and thousands of activities and having involving number of resources per activity.

This research deals with multi resource constrained project scheduling problem. The RCPSP is the main planning and scheduling task for many problems in business and manufacturing system. The details investigation of different problem classes provided insight into the relative advantages of different priority rules.

In this work heuristic based priority rule for the multi-resource with renewable resources are considered. We reviewed three types of heuristic rules: scheduling generation scheme, priority rule and mode selection rule. We have analyzed 90 randomly projects instant with 30 activities, 60 activities and 120 activities. All projects are analyzed by singe pass and multi pass heuristic base priority rules. The computation shows that parallel scheduling method performed better than serial scheduling method with the most of the priority rules tested. Also we have analyzed the effect of the problem parameters such as network complexity (NC), resource factor (RF) and resource strength (RS) on the overall performance in priority rules. The network complexity does not significantly influence the scheduling generation scheme. Resource factor (RF) and resource strength (RS) are highly influence schedule scheme.

6.2 Contribution to Research

In this research, a genetic algorithm procedure has been proposed to produce an optimal and/or near optimal solutions. The permutation based elitist GA shows the performance of search for the optimal and/or suboptimal solution close to the lower bound solutions of RCPSP. The proposed one point crossover selection algorithm makes combination of the individual more randomly to produce best individual for next generation. This randomization process will increase the search space of the fitness function. Hence, the proposed algorithm is searched the fitness value in more wide space and converges between number of generation and standard deviation with a large space. In GA elitist strategy and roulette wheel selection preserved the best solutions to the next generation so as to improve solutions are obtained. The developed algorithm produced several suboptimal solutions to find the best fitness value. The heuristic method cannot generate the optimal solution. Hence, the proposed algorithm for GA is the best meta-heuristic method to search an optimal and/or optimal solution for RCPSP.

A stochastic PERT network with resource constrained model for construction project is presented. Various relationship, time-cost trade-off and resource constrained condition are considered in the model. An optimization algorithm formulation with the objective of minimizing

the project duration is presented. The model allows any linear function for the cost-duration and resource-duration relationships. We have generated a construction model having 40 activities with resource constraint. The proposed algorithm finds cost-duration relationship of the project. The algorithm is capable of handling a wide range of project sizes including large construction projects involving a large number of activities with resource constrained. The proposed model will help the decision-maker to provide a better understanding with the trade-off between the various components that affect the selection, cost, and resource allocation of projects.

6.3 Recommendations for Future Research

The proposed algorithm can be modified and extended to problems where the resources availability varies within the time. The assumption of heuristic methods is based on a static determined execution environment but the project activities must be subject to considerable uncertainty, due to the dynamic and incomplete data. Further research will focus on finding the solution for the multi-project scheduling as well as the resource flow network against the uncertainty.

This study only considered the effect of renewable resource on project scheduling. Future research can consider non-renewable resources. In terms of the issues related to different algorithms, such as the ant colony algorithm, tabu search and simulated annealing comparisons may offer additional choices for practical application. Further work could be conducted to explore the possibility of using activities with multi-mode usage of resources. Furthermore, the genetic algorithm could be used to determine the level of availability of each resource within a certain range of values.

Chapter 7

Bibliography

Boctor, F.F. (1993) “Heuristic for scheduling projects with resource restrictions and several resource-duration modes” *International Journal of production Research*, 31(11), 2547–558.

Boctor, F.F. (1990)“Some efficient multi-heuristic procedures for resource-constrained project scheduling”*European Journal of Operational Research*, 49, 3-13.

Boctor F.F. (1996) “A new and efficient heuristic for scheduling projects with resource restrictions and multiple execution modes” *European Journal of Operational Research,* 90, 349-361.

Boctor F.F.(1996) “Resource-constrained project scheduling by simulated annealing”*International Journal of Production Research,* 34, 2335-2351.

Bell, C.E., Park, K. (1990) "Solving resource-constrained project scheduling problems by A* search", *Naval ResearchLogistics,*37, 61-84.

Bell, C.,Han J. (1991) ” A new heuristic solution method in resource–constrained project scheduling” *Naval Research Logistics*, 38, p 315–331.

Brucker P., Drexl A., MIohring R., Neumann K., Pesch E. (1999) “Resource-constrained project scheduling: notation, classification, models, and methods” *European Journal of Operational Research,* 112(1), 3–41.

Blazewicz, J., Lenstra, J.K., Rinnooy, A.H.G. (1983) "Scheduling subject to resource constraints: Classification and complexity", *Discrete Applied Mathematics* 5, 11-24.

Bean, J. (1994) “Genetic algorithms and random keys for sequencing and optimization” *ORSA Journal on Computing*, 6(2),154–160.

Chan, W.T., Chua, D.K.H. (1996) “Construction resource scheduling with genetic algorithms”*Journal of Construction Engineering and Management*, 122(2), 125–32.

Cho, J. H., Kim, Y. D. (1997) “A Simulated Annealing Algorithm for Resource- Constrained Project Scheduling Problems” *The Journal of the Operational Research Society*, 48(7),736-744.

Christofides, N., Alvarez-Valdes, R., Tamarit, J. M.(1987) “Project Scheduling with Resource Constraints: A Branch and Bound Approach.” *European Journal of Operational Research*, 29, 262-273.

Cooper, D. F. (1976) “Heuristics for Scheduling Resource-Constrained Projects: An Experimental Investigation” *Management Science*, 22(11),1186-1194.

Cooper, D.F. (1977) "A note on serial and parallel heuristics for resource-constrained project scheduling", *Foundations ofControl Engineering* 2, 131-134.

Cho, J.-H., Kim, Y.–D. (1997) “A simulated annealing algorithm for resource constrained project scheduling problems”*Journal of the Operational Research Society,* 48, 735–744.

Chan, W., Chua, D. K. H., Kannan, G. (1996) “Construction Resource Scheduling with Genetic Algorithms ”*Journal of Construction Engineering and Management, ASCE*, 122(2),125-132.

Davis, EW.,(1974) “Network resource allocation ”*Journal of Industrial Engineering* , 4, 22- 32

Davis, EW., and Patterson, J . H.(1975) “A comparison of heuristic and optimum solutions in resource-constrained project scheduling ”*Management Science*, 21, 718- 722.

Davis, EW. (1966) “Resource Allocation in Project Network Models - A Survey” *Journal of Industrial Engineering,* 17, 177-188.

Davis, EW. (1971) “Project Scheduling Under Resource Constraints: Historical Review and Categorization of Procedures”*IlE Transactions,* 5, 297-313.

Davis EW, Heidorn GE.(1971) “An algorithm for optimal project scheduling under multiple resource constraints” *Management Science,* 17, 803–816.

Davis, EW. (1975) “Project Network Summary Measures and Constrained Resource Scheduling, *lIE Transactions,7*, *132-142.*

Drexl, A.(1991) "Scheduling of project networks by job assignment", *Management Science* 37, 1590-1602.

Drexl A. and Grunewald J. (1993) “Non-preemptive multi-mode resource-constrained project scheduling”*IIE Transactions,* 25, 74-81.

Demeulemeester, E. (1995) “Minimizing Resource Availability Costs in Time-Limited Project Networks”*Management Science,* 41, 1590-1598.

Demeulemeester, E. and W. Herroelen(1992) “A Branch-and-Bound Procedure for the Multiple Resource-Constrained Project Scheduling Problem ”*Management Science,* 38, 1803-1818.

Demeulemeester, E. and W. Herroelen(1996) “ An Efficient Optimal Solution Procedure for the Preemptive Resource-Constrained Project Scheduling Problem, *European Journal of Operational Research,* 90, 334-348.

Demeulemeester E.(1995) “Minimizing resource availability costs in time-limited project networks” *Management Science*,41, 1590–1598.

Herroelen W, Demeulemeester E, De Reyck B. (1998) “Resource-constrained project scheduling — a survey of recent developments” *Computers & Operations Research*, 25(4), 279–302.

Hartmann, S. (2001) “Project scheduling with multiple modes: a genetic algorithm. *Annals of Operations Research,* 102, 111–135.

Mohanty R.P., Siddiq M.K.(1989) “Multiple projects multiple resources-constrained scheduling: Some studies”*International Journal of Production Research*, 27(2), 261-280.

Sprecher, A. (2000) “Scheduling Resource-Constrained Projects Competitively at Modest Memory Requirements ”*Management Science*, 46(5), 710-723.

Kolisch, R., Sprecher, A. “PSPLIB - a project scheduling problem library” *European Journal of Operational Research*, 96(1), 1997, p 205–216.

Kolisch, R. “Serial and parallel resource–constrained project scheduling methods revisited: Theory and computation” *European Journal of Operational Research* 90, 1996a, p 320–333.

Kolisch, R. “Efficient priority rules for the resource–constrained project scheduling problem” *Journal of Operations Management,* 14(3), 1996b, p 179–192.

Kolisch, R., and Hartmann, S. “Heuristic Algorithms for the Resource Constrained Project Scheduling Problem: Classification and Computational Analysis.” In: J. Weglarz (Ed.), Project Scheduling-Recent Models, Algorithms and Applications, *Kluwer Academic Publishers*, Boston, 1999, p 147-178.

Kolisch R. “Efficient priority rules for the resource constrained project scheduling problem” *Journal of Operations Management,* 14(3), 1996, p 179–92.

Kim J., Ellis R.D “Permutations-based Elitist genetic algorithm for optimization of large-sized resource constrained project scheduling” *Journal of construction Engineering and Management,* 134(11), 2008, p 904-913.

Lee, J. K., Kim, Y. D. “Search heuristics for resource constrained project scheduling” *Journal of the Operational Research Society,* 47, 1996, p 678-689.

Leu, S., Yang, C. “GA-based multi-criteria optimal model for construction scheduling” *Journal of Construction Engineering and Management* , 125(6),1999, p 420–427.

Leu, S., Yang, C. “GA-Based Multi-criteria Optimal Model for Construction Scheduling.” *Journal of Construction Engineering and Management, ASCE*, 125(6), 1999, p 420-427.

Lova A., Tormos P. “Analysis of Scheduling Schemes and Heuristic Rules Performance in Resource-Constrained Multi-project Scheduling”. *Annals of Operation Research,* 102, 2001, p 263-286

Sprecher, A. (2002) “Network Decomposition Techniques for Resource-Constrained Project Scheduling” *Journal of the Operational Research Society*, 53(4), 405-414.

Talbot, F., Patterson, J. (1979) “Optimal methods for scheduling projects under resource constraints” *Project Management Quarterly*, December, 26–33.

Sprecher, Drexl,. (1998) “Multi-mode resource-constrained project scheduling by a simple, general and powerful sequencing algorithms” *Europe Journal of Operational Research*,107*,* 431-450.

Talbot, F. B., Patterson, J. H(1978) “An Efficient Integer Programming Algorithm with Network Cuts for Solving Resource-Constrained Scheduling Problems” *Management Science*, 24(11), 1163-1174.

Talbot, F.E. (1982) “Resource-Constrained Project Scheduling with Time-Resource Trade-Offs: The Non-preemptive Case” *Management Science,* 28, 1197-1210.

Toklu, Y.C. (2002) “Application of genetic algorithms to construction scheduling with or without resource constraints” *Canadian Journal of Civil Engineering*, 29(3), 421–429.

Thomas, P. R., Salhi, S. (1998) “A Tabu Search Approach for the Resource Constrained Project Scheduling Problem.” *Journal of Heuristics*, 4, 123-139.

Ulusoy, G., Ozdamar, L. (1989) “Heuristic Performance and Network/Resource Characteristics in Resource-Constrained Project Scheduling.” *The Journal of the Operational Research Society*, 40(12), 1145-1152.

Tormos, P., Lova, A.(2001) “A Competitive Heuristic Solution Technique for Resource Constrained Project Scheduling” *Annals of Operations Research*, 102, 65-81.

Thomas P.R., Salhi S. (1997) “An investigation into the relationship of heuristic performance with network-resource characteristics” *Journal of the Operational Research Society,*;48(1), 34–43.

Tormos, P.,Lova, A. (2003) “An Efficient Multi-Pass Heuristic for Project Scheduling with Constrained Resources.” *International Journal of Production Research*, 41(5), 1071-1086.